HYDAC ELECTRONIC

Benutzerhandbuch (Originalanleitung)

User Manual (Translation of original instruction)



Linear-Wegsensoren Serie HLT 1100
mit CANopen Schnittstelle
Linear Transducer Series HLT 1100
with CANopen Interface

- **Zusätzliche Sicherheitshinweise**
- **Installation**
- **Inbetriebnahme**
- **Konfiguration / Parametrierung**
- **Fehlerursachen und Abhilfen**

- ***Additional safety instructions***
- ***Installation***
- ***Commissioning***
- ***Configuration / Parameterization***
- ***Causes of faults and remedies***

- **Software/Support CD:**
  Mat. Nr. 3505546

- ***Software/Support CD:***
  Mat. No. 3505546

Mat. Nr. 669824

Stand: 07.10.2010

Vorwort

Diese Dokumentation muss ständig am Einsatzort verfügbar sein.

Bitte beachten Sie, dass die in dieser Dokumentation gemachten Angaben der Gerätetechnik zu dem Zeitpunkt der Literaturerstellung entsprechen. Abweichungen bei technischen Angaben, Abbildungen und Maßen sind deshalb möglich.

Entdecken Sie beim Lesen dieser Dokumentation Fehler oder haben weitere Anregungen und Hinweise, so wenden Sie sich bitte an:

HYDAC ELECTRONIC GMBH
Technische Dokumentation
Hauptstraße 27
66128 Saarbrücken
-Deutschland-
Tel: +49(0)6897 / 509-01
Fax: +49(0)6897 / 509-1726
Email: electronic@hydac.com

Urheberrechtsschutz

Änderungsvorbehalt

Jegliche Änderungen, die dem technischen Fortschritt dienen, vorbehalten.

Schreibweisen

Kursive oder **fette** Schreibweise steht für den Titel eines Dokuments oder wird zur Hervorhebung benutzt.

`Courier`-Schrift zeigt Text an, der auf dem Bildschirm / Display sichtbar ist und Menüauswahlen von Software.

″ < > ″ weist auf Tasten der Tastatur Ihres Computers hin (wie etwa <RETURN>).

Marken

CANopen ist ein eingetragenes Warenzeichen der CAN in Automation e.V.

Inhaltsverzeichnis

1 Allgemeines

Das vorliegende Benutzerhandbuch beinhaltet folgende Themen:

- Ergänzende Sicherheitshinweise zu den bereits in der Montageanleitung definierten grundlegenden Sicherheitshinweisen
- Elektrische Kenndaten
- Installation
- Inbetriebnahme
- Konfiguration / Parametrierung
- Fehlerursachen und Abhilfen

Da die Dokumentation modular aufgebaut ist, stellt dieses Benutzerhandbuch eine Ergänzung zu anderen Dokumentationen wie z.B. Produktdatenblätter, Maßzeichnungen, Prospekte und der Montageanleitung etc. dar.

Das Benutzerhandbuch kann kundenspezifisch im Lieferumfang enthalten sein, oder kann auch separat angefordert werden.

1.1 Geltungsbereich

Dieses Benutzerhandbuch gilt ausschließlich für folgende Mess-System-Baureihen mit ***CANopen*** Schnittstelle:

- HLT 1100

Die Produkte sind durch aufgeklebte Typenschilder gekennzeichnet und sind Bestandteil einer Anlage.

Es gelten somit zusammen folgende Dokumentationen:

- anlagenspezifische Betriebsanleitungen des Betreibers,
- dieses Benutzerhandbuch,
- und die bei der Lieferung beiliegende Montageanleitung: Mat. Nr. **669822**

1.2 Referenzen

1.	ISO 11898: Straßenfahrzeuge, Austausch von Digitalinformation - Controller Area Network (CAN) für Hochgeschwindigkeits-Kommunikation, November 1993	
2.	Robert Bosch GmbH, CAN-Spezifikation 2.0 Teil A und B, September 1991	
3.	CiA DS-201	V1.1, CAN im OSI Referenz-Model, Februar1996
4.	CiA DS-202-1	V1.1, CMS Service Spezifikation, Februar 1996
5.	CiA DS-202-2	V1.1, CMS Protokoll Spezifikation, Februar 1996
6.	CiA DS-202-3	V1.1, CMS Verschlüsselungsregeln, Februar 1996
7.	CiA DS-203-1	V1.1, NMT Service Spezifikation, Februar 1996
8.	CiA DS-203-2	V1.1, NMT Protokoll Spezifikation, Februar 1996
9.	CiA DS-204-1	V1.1, DBT Service Spezifikation, Februar 1996
10.	CiA DS-204-2	V1.1, DBT Protokoll Spezifikation, Februar 1996
11.	CiA DS-205-1	V1.1, LMT Service Spezifikation, Februar 1996
12.	CiA DS-205-2	V1.1, LMT Protokoll Spezifikation, Februar 1996
13.	CiA DS-206	V1.1, Empfohlene Namenskonventionen für die Schichten, Februar 1996
14.	CiA DS-207	V1.1, Namenskonventionen der Verarbeitungsschichten, Februar 1996
15.	CiA DS-301	V3.0, CANopen Kommunikationsprofil auf CAL basierend, Oktober 1996
16.	CiA DS-406	V2.0, CANopen Profil für Encoder, Mai 1998

1.3 Verwendete Abkürzungen / Begriffe

HLT 1100	Linear-Absolutes-Mess-System für Zylinderintegration
EG	***E***uropäische ***G***emeinschaft
EMV	***E***lektro-***M***agnetische-***V***erträglichkeit
ESD	Elektrostatische Entladung (***E***lectro ***S***tatic ***D***ischarge)
IEC	***I***nternationale ***E***lektrotechnische ***K***ommission
VDE	***V***erein ***D***eutscher ***E***lektrotechniker

CAN-spezifisch

CAL	CAN Application Layer. Die Anwendungsschicht für CAN-basierende Netzwerke ist im CiA-Draft-Standard 201 ... 207 beschrieben.
CAN	Controller Area Network. Datenstrecken-Schicht-Protokoll für serielle Kommunikation, beschrieben in der ISO 11898.
CiA	CAN in Automation. Internationale Anwender- und Hersteller-vereinigung e.V.: gemeinnützige Vereinigung für das Controller Area Network (CAN).
CMS	CAN-based Message Specification. Eines der Serviceelemente in der Anwendungsschicht im CAN Referenz-Model.
COB	Communication Object (CAN Message). Übertragungseinheit im CAN Netzwerk. Daten müssen in einem COB durch das CAN Netzwerk gesendet werden.
COB-ID	COB-Identifier. Eindeutige Zuordnung des COB. Der Identifier bestimmt die Priorität des COB´s im Busverkehr.
DBT	Distributor. Eines der Serviceelemente in der Anwendungsschicht im CAN Referenz-Model. Es liegt in der Verantwortung des DBT´s, COB-ID´s an die COB´s zu verteilen, die von der CMS benutzt werden.
EDS	***E***lectronic-***D***ata-***S***heet (elektronisches Datenblatt)
LSS	Layer Setting Services. Eines der Serviceelemente in der Anwendungsschicht im CAN Referenz-Model. Wird benötigt, um Parameter in den einzelnen Schichten zu konfigurieren.
NMT	Network Management. Eines der Serviceelemente in der Anwendungsschicht im CAN Referenz-Model. Führt die Initialisierung, Konfiguration und Fehlerbehandlung im Busverkehr aus.
PDO	Process Data Object. Objekt für den Datenaustausch zwischen mehreren Geräten.
SDO	Service Data Object. Punkt zu Punkt Kommunikation mit Zugriff auf die Objekt-Datenliste eines Gerätes.



2 Zusätzliche Sicherheitshinweise

2.1 Symbol- und Hinweis-Definition



Warnung
bedeutet, dass Tod, schwere Körperverletzung oder erheblicher Sachschaden eintreten können, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Vorsicht
bedeutet, dass eine leichte Körperverletzung oder ein Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Hinweis
bezeichnet wichtige Informationen bzw. Merkmale und Anwendungstipps des verwendeten Produkts.

2.2 Ergänzende Hinweise zur bestimmungsgemäßen Verwendung

Das Mess-System ist ausgelegt für den Betrieb an CANopen Netzwerken nach dem internationalen Standard ISO/DIS 11898 und 11519-1 bis max. 1 MBaud. Das Profil entspricht dem ***"CANopen Device Profile für Encoder CiA DS-406 V2.0A"***.

Die technischen Richtlinien zum Aufbau des CANopen Netzwerks der CAN-Nutzerorganisation CiA sind für einen sicheren Betrieb zwingend einzuhalten.



Zur bestimmungsgemäßen Verwendung gehört auch:

- ***das*** Beachten aller Hinweise aus diesem Benutzerhandbuch,
- das Beachten der Montageanleitung, insbesondere das dort enthaltene Kapitel ***"Grundlegende Sicherheitshinweise"*** muss vor Arbeitsbeginn gelesen und verstanden worden sein

2.3 Organisatorische Maßnahmen

- Dieses Benutzerhandbuch muss ständig am Einsatzort des Mess-Systems griffbereit aufbewahrt werden.
- Das mit Tätigkeiten am Mess-System beauftragte Personal muss vor Arbeitsbeginn
  - die Montageanleitung, insbesondere das Kapitel ***"Grundlegende Sicherheitshinweise"***,
  - und dieses Benutzerhandbuch, insbesondere das Kapitel "Zusätzliche Sicherheitshinweise",

  gelesen und verstanden haben.

  Dies gilt in besonderem Maße für nur gelegentlich, z.B. bei der Parametrierung des Mess-Systems, tätig werdendes Personal.

3 Elektrische Kenndaten

Versorgungsspannung: .... 12 / 24 V DC

Stromaufnahme ohne Last: .... < 100 mA

Messprinzip: .... magnetostriktiv

Auflösung: .... 0,1 mm

Ausgabedaten: .... Position – 32 Bit Integer
Geschwindigkeit – 16 Bit Integer

CANopen: .... EN 50325-4
Busankopplung: .... ISO 11898-1, ISO 11898-2
CAN Spezifikation 2.0A: .... 11-Bit Identifier
Kommunikations-Profil: .... CiA DS 301 V4.2
Geräte-Profil: .... CiA DS 406 V3.2
LSS: .... CiA DS 305 V2.2
Node-ID: .... 1…127
Baudrate: .... 20, 50, 125, 250, 500, 800, 1000 kbit/s

Anschluss: .... Kupferleitung 4 x 0,5 mm²

Besondere Merkmale: .... Programmierung nachfolgender Parameter
über den CAN-BUS:
- Knotenadresse
- Baudrate
- Zählrichtung
- Presetwert
- PDO mapping

EMV: .... EN 61326/ISO 14982/ISO 7637/ISO 11452/EN 50121



4 CANopen Informationen

CANopen wurde von der CiA entwickelt und ist seit Ende 2002 als europäische Norm EN 50325-4 standardisiert.

CANopen verwendet als Übertragungstechnik die Schichten 1 und 2 des ursprünglich für den Einsatz im Automobil entwickelten CAN-Standards (ISO 11898-2). Diese werden in der Automatisierungstechnik durch die Empfehlungen des CiA Industrieverbandes hinsichtlich der Steckerbelegung, Übertragungsraten erweitert.
Im Bereich der Anwendungsschicht hat CiA den Standard CAL (CAN Application Layer) hervorgebracht.

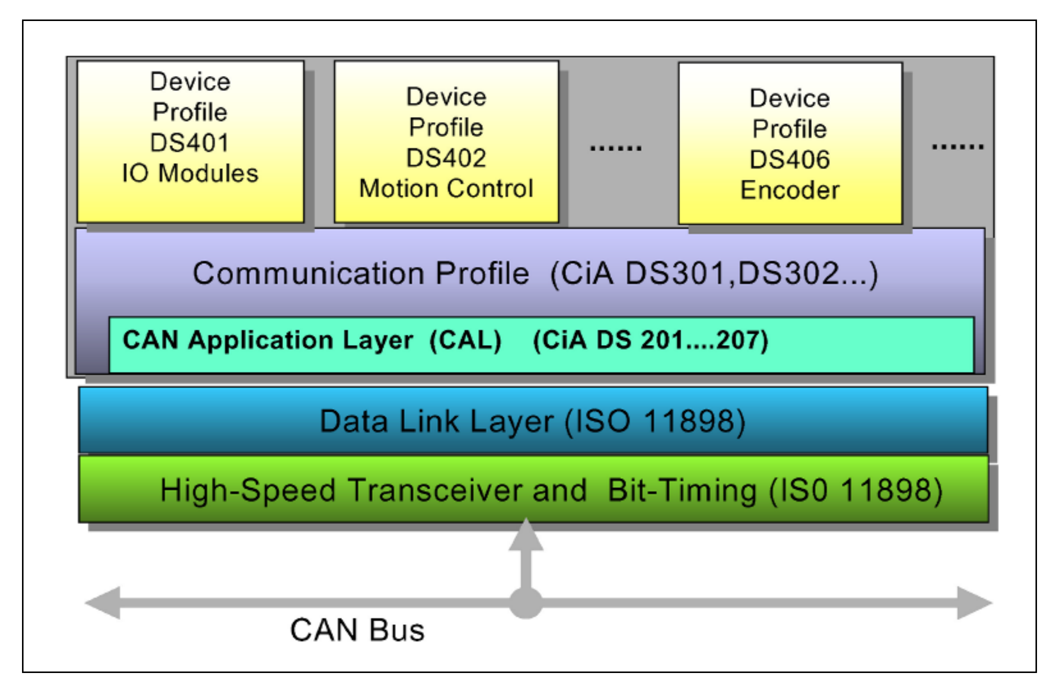


Abbildung 1: CANopen eingeordnet im ISO/OSI-Schichtenmodell

Bei CANopen wurde zunächst das Kommunikationsprofil sowie eine "Bauanleitung" für Geräteprofile entwickelt, in der mit der Struktur des Objektverzeichnisses und den allgemeinen Kodierungsregeln der gemeinsame Nenner aller Geräteprofile definiert ist.

4.1 CANopen – Kommunikationsprofil

Das CANopen Kommunikationsprofil (dokumentiert in CiA DS-301) regelt, wie die Geräte Daten miteinander austauschen. Hierbei werden Echtzeitdaten (z.B. Positionswert) und Parameterdaten (z.B. Zählrichtung) unterschieden. CANopen ordnet diesen, vom Charakter her völlig unterschiedlichen Datenarten, jeweils passende Kommunikationselemente zu.

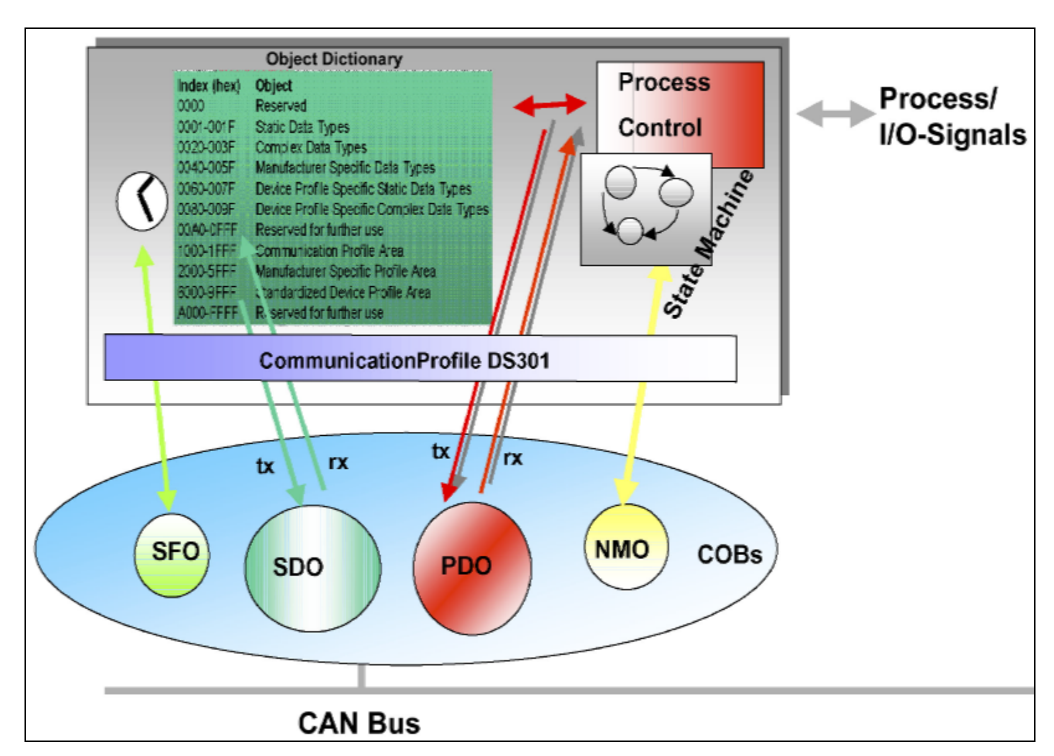


Abbildung 2: Kommunikationsprofil

Special **F**unction **O**bject **(SFO)**

- Synchronization (SYNC)
- Emergency (EMCY) Protokoll

Network **M**anagement **O**bject **(NMO)**

z.B.

- Life / Node-Guarding
- Boot-Up,…
- Error Control Protokoll



4.2 Prozess- und Service-Daten-Objekte

Prozess-Daten-Objekt (PDO)

Prozess-Daten-Objekte managen den Prozessdatenaustausch, wie z.B. die zyklische Übertragung des Positionswertes.
Der Prozessdatenaustausch mit den CANopen PDOs ist "CAN pur", also ohne Protokoll-Overhead. Die Broadcast-Eigenschaften von CAN bleiben voll erhalten. Eine Nachricht kann von allen Teilnehmern gleichzeitig empfangen und ausgewertet werden.
Vom Mess-System werden die beiden Sende-Prozess-Daten-Objekte 1800h für asynchrone (ereignisgesteuerte) Positionsübertragung und 1801h für die synchrone (auf Anforderung) Positionsübertragung verwendet.

Service-Daten-Objekt (SDO)

Service-Daten-Objekte managen den Parameterdatenaustausch, wie z.B. das azyklische Ausführen der Presetfunktion.
Für Parameterdaten beliebiger Größe steht mit dem SDO ein leistungsfähiger Kommunikationsmechanismus zur Verfügung. Hierfür wird zwischen dem Konfigurationsmaster und den angeschlossenen Geräten ein Servicedatenkanal für Parameterkommunikation ausgebildet. Die Geräteparameter können mit einem einzigen Telegramm-Handshake ins Objektverzeichnis der Geräte geschrieben werden bzw. aus diesem ausgelesen werden.

Wichtige Merkmale von SDO und PDO

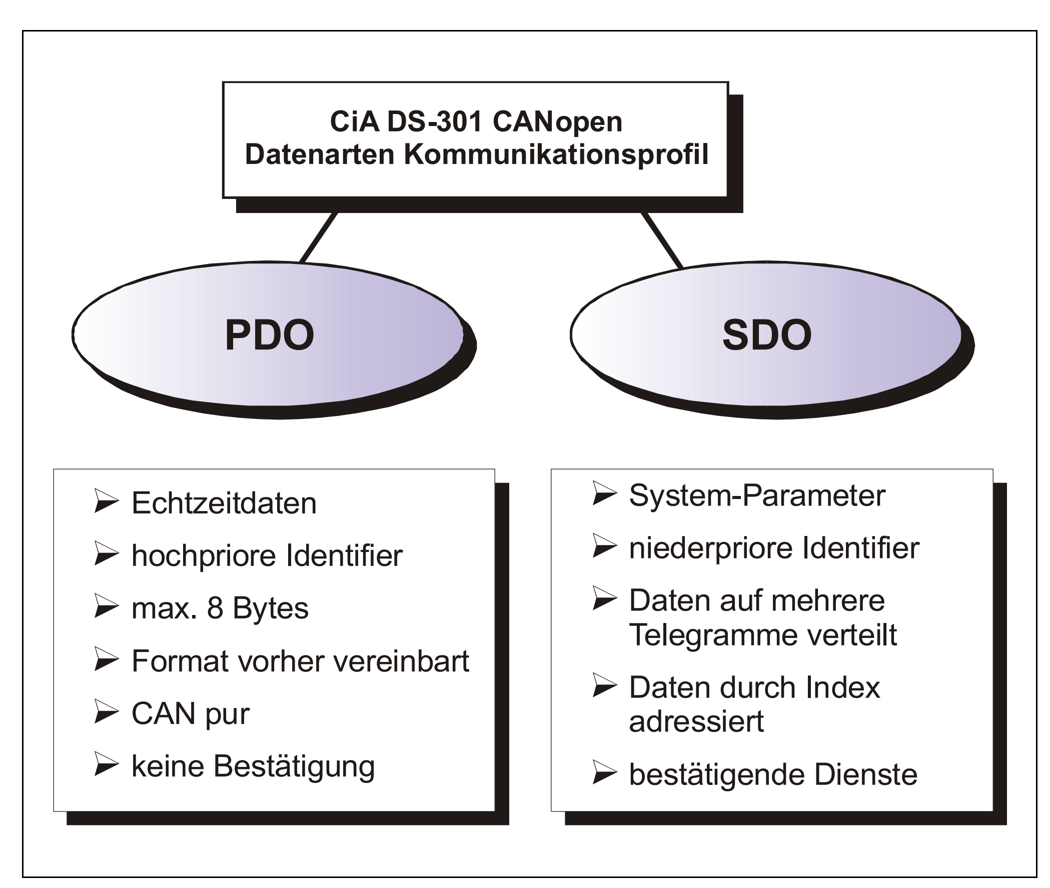


Abbildung 3: Gegenüberstellung von PDO/SDO-Eigenschaften



4.3 Objektverzeichnis (Object Dictionary)

Das Objektverzeichnis strukturiert die Daten eines CANopen-Gerätes in einer übersichtlichen tabellarischen Anordnung. Es enthält sowohl sämtliche Geräteparameter als auch alle aktuellen Prozessdaten, die damit auch über das SDO zugänglich sind.

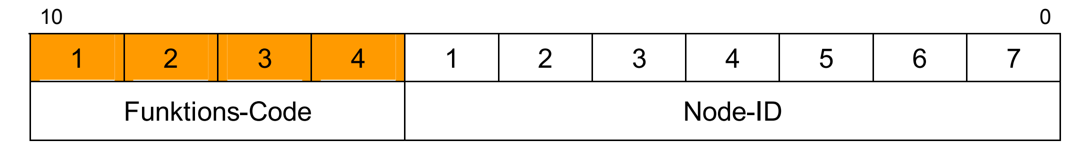
Index	Object	
0000_h	unbenutzt	Standard für alle Geräte
0001_h - $025F_h$	Datentyp Definitionen	
0260_h - $0FFF_h$	Reserviert	
1000_h - $1FFF_h$	Kommunikations-Profilbereich	
2000_h - $5FFF_h$	Herstellerspezifischer-Profilbereich	Geräte-spezifisch
6000_h - $9FFF_h$	Standardisierter-Geräte-Profilbereich	
$A000_h$ - $BFFF_h$	Standardisierter-Schnittstellen-Profilbereich	
$C000_h$ - $FFFF_h$	Reserviert	

Abbildung 4: Aufbau des Objektverzeichnisses

4.4 CANopen Default Identifier, COB-ID

CANopen-Geräte können ohne Konfiguration in ein CANopen–Netzwerk eingesetzt werden. Lediglich die Einstellung einer Busadresse und der Baudrate ist erforderlich. Aus dieser Knotenadresse leitet sich die Identifierzuordnung für die Kommunikationskanäle ab.

COB-Identifier = Funktions-Code + Node-ID



Beispiele

Objekt	Funktions-Code	COB-ID	Index Kommunikations-Parameter
NMT	0000bin	0	–
SYNC	0001bin	80h	1005
PDO1 (tx)	0011bin	181h – 1FFh	1800h

4.5 Übertragung von SDO Nachrichten

Die Übertragung von SDO Nachrichten geschieht über das CMS "Multiplexed-Domain" Protokoll (CIA DS-202-2).
Mit SDOs können Objekte aus dem Objektverzeichnis gelesen oder geschrieben werden. Es handelt sich um einen bestätigten Dienst. Der so genannte ***SDO Client*** spezifiziert in seiner Anforderung „Request“ den Parameter, die Zugriffsart (Lesen/Scheiben) und gegebenenfalls den Wert. Der so genannte ***SDO Server*** führt den Schreib- oder Lesezugriff aus und beantwortet die Anforderung mit einer Antwort „Response“. Im Fehlerfall gibt ein Fehlercode Auskunft über die Fehlerursache. Sende-SDO und Empfangs-SDO werden durch ihre Funktionscodes unterschieden.

Das Mess-System (Slave) entspricht dem SDO Server und verwendet folgende Funktionscodes:

Funktionscode	COB-ID	Bedeutung
`11 (1011 bin)`	`0x580 + Node ID`	**Slave → SDO Client**
`12 (1100 bin)`	`0x600 + Node ID`	**SDO Client → Slave**

Tabelle 1: COB-IDs für Service Data Object (SDO)

4.5.1 SDO-Nachrichtenformat

Der maximal 8 Byte lange Datenbereich einer CAN-Nachricht wird von einem SDO wie folgt belegt:

CCD	Index		Subindex	Daten			
Byte 0	Byte 1, Low	Byte 2, High	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

Tabelle 2: SDO-Nachricht

Der ***Kommando-Code*** (***CCD***) identifiziert bei der SDO Request, ob gelesen oder geschrieben werden soll. Bei einem Schreibauftrag wird zusätzlich die Anzahl der zu schreibenden Bytes im CCD kodiert.
Bei der SDO Response zeigt der CCD an, ob die Request erfolgreich war. Im Falle eines Leseauftrags gibt der CCD zusätzlich Auskunft über die Anzahl der gelesenen Bytes:

CCD	Bedeutung	Gültig für
0x23	4 Byte schreiben	SDO Request
0x2B	2 Byte schreiben	SDO Request
0x2F	1 Byte schreiben	SDO Request
0x60	Schreiben erfolgreich	SDO Response
0x80	Fehler	SDO Response
0x40	Leseanforderung	SDO Request
0x43	4 Byte Daten gelesen	SDO Response auf Leseanforderung
0x4B	2 Byte Daten gelesen	SDO Response auf Leseanforderung
0x4F	1 Byte Daten gelesen	SDO Response auf Leseanforderung

Tabelle 3: Kommando-Codes für SDO

Im Fall eines Fehlers (SDO Response CCD = 0x80) enthält der Datenbereich einen 4-Byte-Fehlercode, der über die Fehlerursache Auskunft gibt. Die Bedeutung der Fehlercodes ist aus der Tabelle 10, Seite 60 zu entnehmen.

Segment Protokoll, Datensegmentierung

Manche Objekte beinhalten Daten, die größer als 4 Byte sind. Um diese Daten lesen zu können, muss das „Segment Protokoll“ benutzt werden.
Zunächst wird der Lesevorgang wie ein gewöhnlicher SDO-Dienst mit dem Kommando-Code = 0x40 eingeleitet. Über die Response wird angezeigt, um wie viele Datensegmente es sich handelt und wie viele Bytes gelesen werden können. Mit nachfolgenden Leseanforderungen können dann die einzelnen Datensegmente gelesen werden. Ein Datensegment besteht jeweils aus 7 Bytes.

Beispiel für das Lesen eines Datensegmentes:

Telegramm 1

CCD	Bedeutung	Gültig für
0x40	Leseanforderung, Einleitung	SDO Request
0x41	1 Datensegment vorhanden Die Anzahl der zu lesenden Bytes steht in den Bytes 4 bis 7.	SDO Response

Telegramm 2

CCD	Bedeutung	Gültig für
0x60	Leseanforderung	SDO Request
0x01	Kein weiteres Datensegment vorhanden. Die Bytes 1 bis 7 beinhalten die angeforderten Daten.	SDO Response

4.5.2 Lese SDO

„Domain Upload“ einleiten

Anforderungs-Protokoll-Format:

COB-Identifier = 600h + Node-ID

Lese SDO´s								
Byte	0	1	2	3	4	5	6	7
Inhalt	Code	Index		Sub-index	Daten 0	Daten 1	Daten 2	Daten 3
	40h	Low	High	Byte	0	0	0	0

Das „Lese-SDO“ Telegramm muss an den Slave gesendet werden.

Der Slave antwortet mit folgendem Telegramm:

Antwort-Protokoll-Format:

COB-Identifier = 580h + Node-ID

Lese SDO´s								
Byte	0	1	2	3	4	5	6	7
Inhalt	Code	Index		Sub-index	Daten 0	Daten 1	Daten 2	Daten 3
	4xh	Low	High	Byte	Daten	Daten	Daten	Daten

Format-Byte 0:
MSB LSB

7	6	5	4	3	2	1	0
0	1	0	0	n		1	1

n = Anzahl der Datenbytes (Bytes 4-7), welche keine Daten beinhalten.

Wenn nur 1 Datenbyte (Daten 0) Daten enthält, ist der Wert von Byte 0 = "4Fh".

Ist Byte 0 = 80h, wird die Übertragung abgebrochen.

4.5.3 Schreibe SDO

„Domain Download“ einleiten

Anforderungs-Protokoll-Format:

COB-Identifier = 600h + Node-ID

Schreibe SDO´s								
Byte	0	1	2	3	4	5	6	7
Inhalt	Code	Index		Sub-index	Daten 0	Daten 1	Daten 2	Daten 3
	2xh	Low	High	Byte	0	0	0	0

Format-Byte 0:
MSB LSB

7	6	5	4	3	2	1	0
0	0	1	0	n		1	1

n = Anzahl der Datenbytes (Bytes 4-7), welche keine Daten beinhalten.

Wenn nur 1 Datenbyte (Daten 0) Daten enthält, ist der Wert von Byte 0 = "2Fh".

Das „Schreibe-SDO“ Telegramm muss an den Slave gesendet werden.

Der Slave antwortet mit folgendem Telegramm:

Antwort-Protokoll-Format:

COB-Identifier = 580h + Node-ID

Lese SDO´s								
Byte	0	1	2	3	4	5	6	7
Inhalt	Code	Index		Sub-index	Daten 0	Daten 1	Daten 2	Daten 3
	60h	Low	High	Byte	0	0	0	0

Ist Byte 0 = 80h, wird die Übertragung abgebrochen.

4.6 Netzwerkmanagement, NMT

Das Netzwerkmanagement unterstützt einen vereinfachten Hochlauf (Boot-Up) des Netzes. Mit einem einzigen Telegramm lassen sich z.B. alle Geräte in den Betriebszustand (Operational) versetzen.

Das Mess-System befindet sich nach dem Einschalten zunächst im "Vor-Betriebszustand", (2).

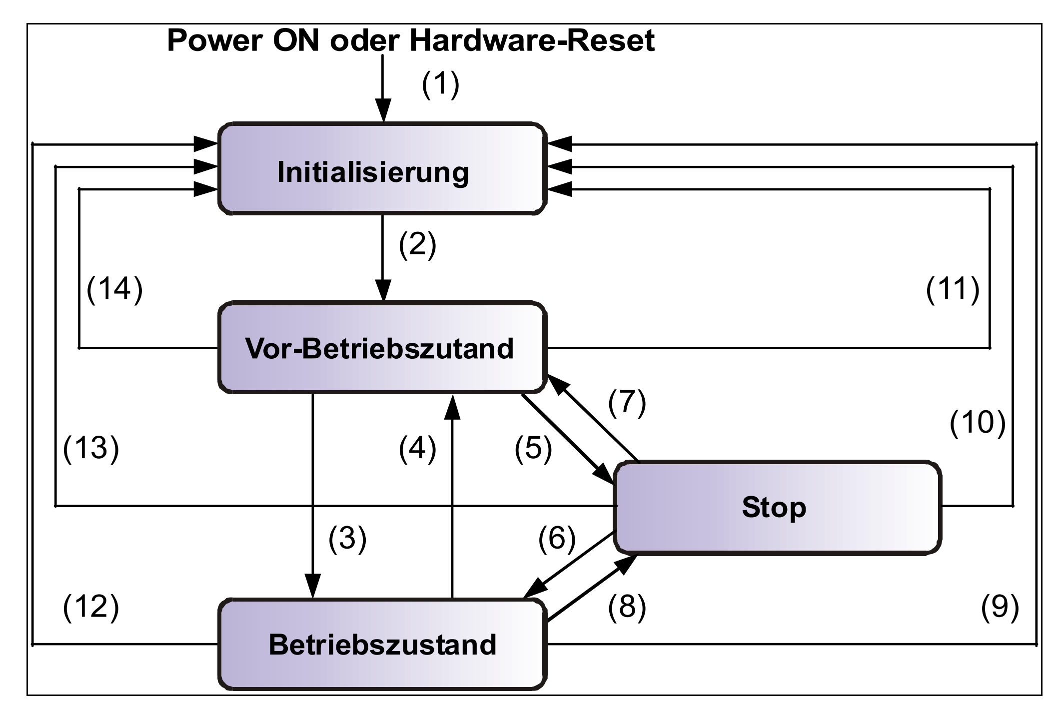


Abbildung 5: Boot-Up-Mechanismus des Netzwerkmanagements

Zustand	Beschreibung
(1)	Automatische Initialisierung nach dem Einschalten
(2)	Beendigung der Initialisierung --> Vor-Betriebszustand
(3),(6)	Start_Remote_Node --> Betriebszustand
(4),(7)	Enter_PRE-OPERATIONAL_State --> Vor-Betriebszustand
(5),(8)	Stop_Remote_Node --> Stop
(9),(10),(11)	Reset_Node --> Reset Knoten
(12),(13),(14)	Reset_Communication --> Reset Kommunikation

4.6.1 Netzwerkmanagement-Dienste

Das ***Network Management*** (***NMT***) hat die Aufgabe, Teilnehmer eines CANopen-Netzwerks zu initialisieren, die Teilnehmer in das Netz aufzunehmen, zu stoppen und zu überwachen.

NMT-Dienste werden von einem ***NMT-Master*** initiiert, der einzelne Teilnehmer (***NMT-Slave***) über deren Node ID anspricht. Eine NMT-Nachricht mit der Node ID 0 richtet sich an **alle** NMT-Slaves.

Das Mess-System entspricht einem NMT-Slave.

4.6.1.1 NMT-Dienste zur Gerätekontrolle

Die NMT-Dienste zur Gerätekontrolle verwenden die **COB-ID 0** und erhalten so die höchste Priorität.

Vom Datenfeld der CAN-Nachricht werden nur die ersten beiden Byte verwendet:

CCD	Node ID
Byte 0	Byte 1

Folgende Kommandos sind definiert:

CCD	Bedeutung	Zustand
-	Automatische Initialisierung nach dem Einschalten	(1)
-	Beendigung der Initialisierung --> PRE-OPERATIONAL	(2)
0x01	**Start Remote Node** Teilnehmer soll in den Zustand OPERATIONAL wechseln und damit den normalen Netzbetrieb starten	(3),(6)
0x02	**Stop Remote Node** Teilnehmer soll in den Zustand STOPPED übergehen und damit seine Kommunikation stoppen. Eine aktive Verbindungsüberwachung bleibt aktiv.	(5),(8)
0x80	**Enter PRE-OPERATIONAL** Teilnehmer soll in den Zustand PRE-OPERATIONAL gehen. Alle Nachrichten außer PDOs können verwendet werden.	(4),(7)
0x81	**Reset Node** Werte der Profilparameter des Objekts auf Default-Werte setzen. Danach Übergang in den Zustand RESET COMMUNICATION.	(9),(10), (11)
0x82	**Reset Communication** Teilnehmer soll in den Zustand RESET COMMUNICATION gehen. Danach Übergang in den Zustand INITIALIZATION, erster Zustand nach dem Einschalten.	(12),(13), (14)

Tabelle 4: NMT-Dienste zur Gerätekontrolle

4.6.1.2 NMT-Dienste zur Verbindungsüberwachung

Mit der Verbindungsüberwachung kann ein NMT-Master den Ausfall eines NMT-Slave und/oder ein NMT-Slave den Ausfall des NMT-Master erkennen:

- **Node Guarding und Life Guarding:**
  Mit diesen Diensten überwacht ein NMT-Master einen NMT-Slave

Das **Node Guarding** wird dadurch realisiert, dass der NMT-Master in regelmäßigen Abständen den Zustand eines NMT-Slave anfordert. Das Toggle-Bit 2^7 im „Node Guarding Protocol“ toggelt nach jeder Abfrage:

Beispiel:
0x85, 0x05, 0x85 … --> kein Fehler
0x85, 0x05, 0x05 … --> Fehler

Ist zusätzlich das **Life Guarding** aktiv, erwartet der NMT-Slave innerhalb eines bestimmten Zeitintervalls eine derartige Zustandsabfrage durch den NMT-Master. Ist dies nicht der Fall, wechselt der Slave in den PRE-OPERATIONAL Zustand.

Die NMT-Dienste zur Verbindungsüberwachung verwenden den Funktionscode `1110 bin`, also die **COB-ID 0x700+Node ID**.

<table>
<tr><th>Index</th><th colspan="2">Beschreibung</th></tr>
<tr><td>0x100C</td><td>Guard Time [ms]</td><td rowspan="2">Spätestens nach Ablauf des Zeitintervalls
Life Time = Guard Time x Life Time Factor [ms]
erwartet der NMT-Slave eine Zustandsabfrage durch den Master.

Ist die Guard Time = 0, wird der entsprechende NMT-Slave nicht vom Master überwacht.
Ist die Life Time = 0, ist das Life Guarding abgeschaltet.</td></tr>
<tr><td>0x100D</td><td>Life Time Factor</td></tr>
</table>

Tabelle 5: Parameter für NMT-Dienste

4.7 Layer setting services (LSS) und Protokolle

Die LSS-Dienste und Protokolle, dokumentiert in CiA DS-305 V2.2, unterstützen das Abfragen und Konfigurieren verschiedener Parameter des Data Link Layers und des Application Layers eines LSS-Slaves durch ein LSS-Master über das CAN Netzwerk.

Unterstützt werden folgende Parameter:

- Node-ID
- Baudrate
- LSS-Adresse, gemäß dem Identity Objekt 1018h

Der Zugriff auf den LSS-Slave erfolgt dabei über seine LSS-Adresse, bestehend aus:

- Vendor-ID
- Produkt-Code
- Revisions-Nummer und
- Serien-Nummer

Das Mess-System unterstützt folgende Dienste:

Switch mode services

- Switch mode selective
  - einen bestimmten LSS-Slave ansprechen
- Switch mode global
  - alle LSS-Slaves ansprechen

Configuration services

- Configure Node-ID
  - Node-ID konfigurieren
- Configure bit timing parameters
  - Baudrate konfigurieren
- Activate bit timing parameters
  - Baudrate aktivieren
- Store configured parameters
  - konfigurierte Parameter speichern

Inquiry services

- Inquire LSS-address
  - LSS-Adresse anfragen
- Inquire Node-ID
  - Node-ID anfragen

Identification services

- LSS identify remote slave
  - Identifizierung von LSS-Slaves innerhalb eines bestimmten Bereichs
- LSS identify slave
  - Rückmeldung der LSS-Slaves auf das vorherige Kommando
- LSS identify non-configured remote slave
  - Identifizierung von nicht-konfigurierten LSS-Slaves, Node-ID = FFh
- LSS identify non-configured slave
  - Rückmeldung der LSS-Slaves auf das vorherige Kommando

4.7.1 LSS-Modes und Dienste

Der LSS entspricht einer Zustandsmaschine und definiert das Verhalten eines LSS-Slaves.Gesteuert wird die Zustandsmaschine durch LSS COBs erzeugt durch einen LSS-Master, oder NMT COBs erzeugt durch einen NMT-Master, oder lokale NMTZustandsübergänge.

Die LSS-Modes unterstützen folgende Zustände:

(0) Initial: Pseudo-Zustand, zeigt die Aktivierung des FSAs an
(1) LSS waiting: Unterstützung aller Dienste wie unten angegeben
(2) LSS configuration: Unterstützung aller Dienste wie unten angegeben
(3) Final: Pseudo-Zustand, zeigt die Deaktivierung des FSAs an

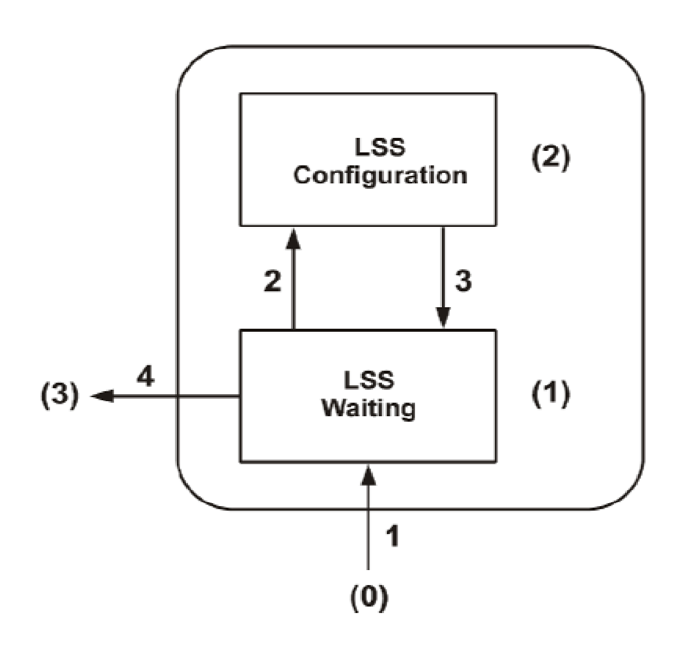


Abbildung 6: LSS-Modes

Zustandsverhalten der unterstützten Dienste

Dienste	Waiting	Configuration
Switch mode global	Ja	Ja
Switch mode selective	Ja	Nein
Activate bit timing parameters	Nein	Ja
Configure bit timing parameters	Nein	Ja
Configure Node-Id	Nein	Ja
Store configuration	Nein	Ja
Inquire LSS-address	Nein	Ja
LSS identify remote slave	Ja	Ja
LSS identify slave	Ja	Ja
LSS identify non-configuration remote slave	Ja	Ja
LSS identify non-configured slave	Ja	Ja

LSS FSA Zustandsübergänge

Übergang	Ereignisse	Aktionen
1	Automatischer Übergang nach der Initialisierung beim Eintritt entweder in den NMT PRE OPERATIONAL Zustand oder NMT STOPPED Zustand, oder NMT RESET COMMUNICATION Zustand mit Node-ID = FFh.	keine
2	LSS 'switch state global' Kommando mit Parameter 'configuration_switch' oder 'switch state selective' Kommando	keine
3	LSS 'switch state global' Kommando mit Parameter 'waiting_switch'	keine
4	Automatischer Übergang, wenn eine ungültige Node-ID geändert wurde und die neue Node-ID erfolgreich im nichtflüchtigen Speicher abgelegt werden konnte UND der Zustand LSS waiting angefordert wurde.	keine

Sobald das LSS FSA weitere Zustandsübergänge im NMT FSA von NMT PRE OPERATIONAL auf NMT STOPPED und umgekehrt erfährt, führt dies nicht zum Wiedereintritt in den LSS FSA.

4.7.2 Übertragung von LSS-Diensten

Über die LSS-Dienste fordert der LSS-Master die einzelnen Dienste an, welche dann durch den LSS-Slave ausgeführt werden. Die Kommunikation zwischen LSS-Master und LSS-Slave wird über die implementierten LSS-Protokolle vorgenommen.
Ähnlich wie bei der SDO-Übertragung, werden auch hier zwei COB-IDs für das Senden und Empfangen benutzt:

COB-ID	Bedeutung
0x7E4	**LSS-Slave → LSS-Master**
0x7E5	**LSS-Master → LSS-Slave**

Tabelle 6: COB-IDs für LSS Services

4.7.2.1 LSS-Nachrichtenformat

Der maximal 8 Byte lange Datenbereich einer CAN-Nachricht wird von einem LSS-Dienst wie folgt belegt:

CS	Daten						
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

Tabelle 7: LSS-Nachricht

Byte 0 enthält die ***Command-Specifier*** (CS), danach folgen 7 Byte für die Daten.

4.7.3 Switch mode Protokolle

4.7.3.1 Switch mode global Protokoll

Das angegebene Protokoll hat den *Switch mode global service* implementiert und steuert das Zustandsverhalten des LSS-Slaves. Über den LSS-Master können alle LSS-Slaves im Netzwerk in den *Waiting Mode* oder *Configuration Mode* gebracht werden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Mode**	Reserved by CiA					
0x7E5	04	0 = Waiting Mode 1 = Configuration Mode						

4.7.3.2 Switch mode selective Protokoll

Das angegebene Protokoll hat den `Switch mode selective service` implementiert und steuert das Zustandsverhalten des LSS-Slaves. Über den LSS-Master kann nur der LSS-Slave im Netzwerk in den `Configuration Mode` gebracht werden, dessen LSS- Adressattribute der LSS-Adresse entsprechen.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Vendor-Id**				Reserved by CiA		
0x7E5	64	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Product-Code**				Reserved by CiA		
0x7E5	65	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number**				Reserved by CiA		
0x7E5	66	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number**				Reserved by CiA		
0x7E5	67	LSB			MSB			

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	68							

Error Code
- 0: Ausführung erfolgreich
- 1…254: Reserved
- 255: Applikationsspezifischer Fehler aufgetreten

Specific Error

Wenn Error Code = 255 --> applikationsspezifischer Fehler aufgetreten, sonst reserviert durch die CiA

4.7.4 Configuration Protokolle

4.7.4.1 Configure Node-Id Protokoll

Das angegebene Protokoll hat den *Configure NMT-Address service* implementiert. Über den LSS-Master kann die Node-ID eines einzelnen LSS-Slaves im Netzwerk konfiguriert werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden. Zur Speicherung der neuen Node-ID muss das *Store configuration protocol* an den LSS-Slave übertragen werden. Um die neue Node-ID zu aktivieren, muss der NMT-Dienst *Reset Communication* (0x82) aufgerufen werden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Node-ID**	Reserved by CiA					
0x7E5	17	1…127 und 255						

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Error Code**	**Spec. Error**	Reserved by CiA				
0x7E4	17							

Error Code
- 0: Ausführung erfolgreich
- 1…254: Reserved
- 255: applikationsspezifischer Fehler aufgetreten

Specific Error

Wenn Error Code = 255 --> applikationsspezifischer Fehler aufgetreten,
sonst reserviert durch die CiA

4.7.4.2 Configure bit timing parameters Protokoll

Das angegebene Protokoll hat den `Configure bit timing parameters service` implementiert. Über den LSS-Master kann die Baudrate eines einzelnen oder aller LSS-Slaves im Netzwerk konfiguriert werden. Zur Speicherung der neuen Baudrate muss das `Store configuration protocol` an den LSS-Slave übertragen werden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Table Selector**	**Table Index**	Reserved by CiA				
0x7E5	19	0	0…7					

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Error Code**	**Spec. Error**	Reserved by CiA				
0x7E4	19							

Table Selector
0: Standard CiA Baudraten-Tabelle

Table Index
0: 1 Mbit/s
1: 800 kbit/s
2: 500 kbit/s
3: 250 kbit/s
4: 125 kbit/s
5: reserved
6: 50 kbit/s
7: 20 kbit/s

Error Code
0: Ausführung erfolgreich
1: selektierte Baudrate nicht unterstützt
2…254: Reserved
255: applikationsspezifischer Fehler aufgetreten

Specific Error
Wenn Error Code = 255 --> applikationsspezifischer Fehler aufgetreten,
sonst reserviert durch die CiA

4.7.4.3 Activate bit timing parameters Protokoll

Das angegebene Protokoll hat den `Activate bit timing parameters service` implementiert und aktiviert die über `Configure bit timing parameters protocol` festgelegte Baudrate bei allen LSS-Slaves im Netzwerk, die sich im `Configuration Mode` befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Switch Delay [ms]**		Reserved by CiA				
0x7E5	21	LSB	MSB					

Switch Delay
Der Parameter `Switch Delay` definiert die Länge zweier Verzögerungsperioden (D1, D2) mit gleicher Länge. Damit wird das Betreiben des Busses mit unterschiedlichen Baudratenparametern verhindert.
Nach Ablauf der Zeit D1 und einer individuellen Verarbeitungsdauer wird die Umschaltung intern im LSS-Slave vorgenommen. Nach Ablauf der Zeit D2 meldet sich der LSS-Slave wieder mit CAN-Nachrichten und der neu eingestellten Baudrate.

Es gilt:
Switch Delay > längste vorkommende Verarbeitungsdauer eines LSS-Slaves

4.7.4.4 Store configuration Protokoll

Das angegebene Protokoll hat den `Store configured parameters service` implementiert. Über den LSS-Master können die konfigurierten Parameter eines einzelnen LSS-Slaves im Netzwerk in den nichtflüchtigen Speicher abgelegt werden. Hierbei darf sich nur ein LSS-Slave im `Configuration Mode` befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	23							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Error Code**	**Spec. Error**	Reserved by CiA				
0x7E4	23							

Error Code
0: Ausführung erfolgreich
1: `Store configuration` nicht unterstützt
2…254: Reserved
255: applikationsspezifischer Fehler aufgetreten

Specific Error
Wenn Error Code = 255 --> applikationsspezifischer Fehler aufgetreten,
sonst reserviert durch die CiA



4.7.5 Inquire LSS-Address Protokolle

4.7.5.1 Inquire Identity Vendor-ID Protokoll

Das angegebene Protokoll hat den `Inquire LSS-Address service` implementiert. Über den LSS-Master kann die Vendor-ID eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im `Configuration Mode` befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	90							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Vendor-Id** (= Index 1018h:01)				Reserved by CiA		
0x7E4	90	LSB			MSB			

4.7.5.2 Inquire Identity Product-Code Protokoll

Das angegebene Protokoll hat den `Inquire LSS-Address service` implementiert. Über den LSS-Master kann der Hersteller-Gerätename eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im `Configuration Mode` befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	91							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Product-Code** (= Index 1018h:02)				Reserved by CiA		
0x7E4	91	LSB			MSB			

4.7.5.3 Inquire Identity Revision-Number Protokoll

Das angegebene Protokoll hat den *Inquire LSS-Address service* implementiert. Über den LSS-Master kann die Revisionsnummer eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	92							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number** (= Index 1018h:03)				Reserved by CiA		
0x7E4	92	LSB			MSB			

4.7.5.4 Inquire Identity Serial-Number Protokoll

Das angegebene Protokoll hat den *Inquire LSS-Address service* implementiert. Über den LSS-Master kann die Seriennummer eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im *Configuration Mode* befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	93							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number** (= Index 1018h:04)				Reserved by CiA		
0x7E4	93	LSB			MSB			

4.7.5.5 Inquire Node-ID Protokoll

Das angegebene Protokoll hat `den Inquire Node-ID service` implementiert. Über den LSS-Master kann die Node-ID eines einzelnen LSS-Slaves im Netzwerk ausgelesen werden. Hierbei darf sich nur ein LSS-Slave im `Configuration Mode` befinden.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	94							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Node-ID**	Reserved by CiA					
0x7E4	94	1…127 und 255	MSB					

Node-ID
Entspricht der Node-ID des selektierten Gerätes. Wenn die Node-ID eben gerade erst über den `Configure Node-ID service` geändert wurde, wird die ursprüngliche Node-ID zurückgemeldet. Erst nach Ausführung des NMT-Dienstes Reset `Communication (0x82)` wird die aktuelle Node-ID zurückgemeldet.

4.7.6 Identification Protokolle

4.7.6.1 LSS identify remote slave Protokoll

Das angegebene Protokoll hat den *LSS identify remote slaves service* implementiert. Über den LSS-Master können LSS-Slaves im Netzwerk in einem bestimmten Bereich identifiziert werden. Alle LSS-Slaves, die dem angegebenen Vendor-ID, Product-Code, Revision-No. und Serial-No. – Bereich entsprechen, antworten mit dem *LSS identify slave protocol*.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Vendor-Id**				Reserved by CiA		
0x7E5	70	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Product-Code**				Reserved by CiA		
0x7E5	71	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number-Low**				Reserved by CiA		
0x7E5	72	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number-High**				Reserved by CiA		
0x7E5	73	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number-Low**				Reserved by CiA		
0x7E5	74	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number-High**				Reserved by CiA		
0x7E5	75	LSB			MSB			

4.7.6.2 LSS identify slave Protokoll

Das angegebene Protokoll hat den `LSS identify slave service` implementiert. Alle LSS-Slaves, die den im `LSS identify remote slaves protocol` angegebenen LSS-Adress-Attributen entsprechen, antworten mit diesem Protokoll.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	79							

4.7.6.3 LSS identify non-configured remote slave Protokoll

Das angegebene Protokoll hat den `LSS identify non-configured remote slave service` implementiert. Über den LSS-Master werden alle nichtkonfigurierten LSS-Slaves (NodeID = FFh) im Netzwerk identifiziert. Die betreffenden LSS-Slaves antworten mit dem `LSS identify non-configured remote slave protocol.`

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	76							

4.7.6.4 LSS identify non-configured slave Protokoll

Das angegebene Protokoll hat den `LSS identify non-configured slave service` implementiert. Alle LSS-Slaves, die eine ungültige Node-ID (FFh) besitzen, antworten nach Ausführung des `LSS identify non-configured slave protocol` mit diesem Protokoll.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	80							

4.7.6.5 LSS fastscan Protokoll

Das angegebene Protokoll hat `den LSS fastscan service` implementiert. Die LSS-Slaves, die über alle 4 LSS Sub-Einträge identifiziert wurden wechseln danach in den `Configuration Mode`.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**ID Nummer**				**Bit Check**	**LSS Sub**	**LSS Next**
0x7E4	81	LSB			MSB			

4.8 Geräteprofil

Die CANopen Geräteprofile beschreiben das "was" der Kommunikation. In ihnen wird die Bedeutung der übertragenen Daten eindeutig und hersteller-unabhängig festgelegt. So lassen sich die Grundfunktionen einer jeden Geräteklasse

z.B. für Encoder: ***CiA DS-406***

einheitlich ansprechen. Auf der Grundlage dieser standardisierten Profile kann auf identische Art und Weise über den Bus auf CANopen Geräte zugegriffen werden. Damit sind Geräte, die dem gleichen Geräteprofil folgen, weitgehend untereinander austauschbar.

Weitere Informationen zum CANopen erhalten Sie auf Anfrage von der ***CAN in Automation*** Nutzer- und Herstellervereinigung (CiA) unter nachstehender Adresse:

CAN in Automation
Am Weichselgarten 26
DE-91058 Erlangen

Tel. +49-9131-69086-0
Fax +49-9131-69086-79

Website: www.can-cia.org
e-mail: headquarters@can-cia.org

---

5 Installation / Inbetriebnahmevorbereitung

Das CANopen System wird in Bustopologie mit Abschlusswiderständen (120 Ohm) am Anfang und am Ende verkabelt. Stichleitungen sollten möglichst vermieden werden. Das Kabel ist als geschirmtes Twisted Pair Kabel auszuführen und sollte eine Impedanz von 120 Ohm und einen Widerstand von 70 mΩ/m haben. Die Datenübertragung erfolgt über die Signale CAN-H und CAN-L mit einem gemeinsamen GND als Datenbezugspotential. Optional kann auch die Versorgungsspannung mitgeführt werden.

In einem CANopen Netzwerk können maximal **127** Teilnehmer angeschlossen werden. Das Mess-System unterstützt den Node-ID Bereich von 1–127. Die Übertragungsgeschwindigkeit lässt sich per LSS-Protokoll einstellen und unterstützt die Baudraten 20 kbit/s, 125 kbit/s, 500 kbit/s und 1 Mbit/s.

Die Länge eines CANopen Netzwerkes ist abhängig von der Übertragungsgeschwindigkeit und ist nachfolgend dargestellt:

Kabelquerschnitt	20 kbit/s	125 kbit/s	500 kbit/s	1 Mbit/s
0.25 mm^2 – 0.34 mm^2	2500 m	500 m	100 m	25 m



Um einen sicheren und störungsfreien Betrieb zu gewährleisten, sind die

- *ISO 11898,*
- *die Empfehlungen der CiA DR 303-1*
  (CANopen cabling and connector pin assignment)
- *und sonstige einschlägige Normen und Richtlinien zu beachten!*

Insbesondere sind die EMV-Richtlinie sowie die Schirmungs- und Erdungsrichtlinien in den jeweils gültigen Fassungen zu beachten!

5.1 Anschluss

Der Anschluss kann mit Hilfe der beigelegten Geräte-spezifischen Steckerbelegung durchgeführt werden.

Für die Versorgung sind paarweise verdrillte und geschirmte Kabel zu verwenden !

5.2 Einschalten der Versorgungsspannung

Nachdem der Anschluss und alle Einstellungen vorgenommen worden sind, kann die Versorgungsspannung eingeschaltet werden.

Nach dem Einschalten der Versorgungsspannung und Beendigung der Initialisierung geht das Mess-System in den Vor-Betriebszustand (PRE-OPERATIONAL). Dieser Zustand wird durch die Boot-Up-Meldung „**COB-ID 0x700+Node ID**" bestätigt. Falls das Mess-System einen internen Fehler erkennt, wird eine Emergency-Meldung mit dem Fehlercode übertragen
(siehe Kapitel „Emergency-Meldung", Seite 57).

Im PRE-OPERATIONAL-Zustand ist zunächst nur eine Parametrierung über Service-Daten-Objekte möglich. Es ist aber möglich, PDOs unter Nutzung von SDOs zu konfigurieren. Ist das Mess-System in den Zustand OPERATIONAL überführt worden, ist auch eine Übertragung von PDOs möglich.

5.3 Einstellen der Node-ID und Baudrate mittels LSS-Dienste

5.3.1 Konfiguration der Node-ID, Ablauf

Annahme:

- LSS-Adresse unbekannt
- der LSS-Slave ist der einzige Teilnehmer in Netzwerk
- es soll die Node-ID 12 dez. eingestellt werden

Vorgehensweise:

- LSS-Slave mit dem Dienst 04 *Switch mode global protocol*,
  Mode = 1 in den *Configuration Mode* bringen.
- Dienst 17 *Configure NMT-Address protocol*, Node-ID = 12 ausführen.
  --> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,
  --> Error Code = 0.
- Dienst 23 *Store configuration protocol* ausführen.
  --> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,
  --> Error Code = 0.
- Versorgungsspannung des LSS-Slaves aus-, danach wieder einschalten. Die neue Konfiguration ist jetzt aktiv.

5.3.2 Konfiguration der Baudrate, Ablauf

Annahme:

- LSS-Adresse unbekannt
- der LSS-Slave ist der einzige Teilnehmer in Netzwerk
- es soll die Baudrate 125 kbit/s eingestellt werden

Vorgehensweise:

- LSS-Slave mit dem Dienst 04 *Switch mode global protocol*,
  Mode = 1 in den *Configuration Mode* bringen.
- Dienst 19 *Configure bit timing parameters protocol* ausführen,
  Table Selector = 0, Table Index = 4
  --> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,
  --> Error Code = 0.
- Dienst 21 *Activate bit timing parameters protocol* aufrufen,
  damit die neue Baudrate aktiv wird.
- Dienst 23 *Store configuration protocol* ausführen.
  --> Rückmeldung abwarten und erfolgreiche Ausführung überprüfen,
  --> Error Code = 0.
- Versorgungsspannung des LSS-Slaves aus-, danach wieder einschalten. Die neue Konfiguration ist jetzt aktiv.

6 Inbetriebnahme

6.1 CAN – Schnittstelle

Die CAN-Bus-Schnittstelle ist durch die internationale Norm ISO/DIS 11898 definiert und spezifiziert die zwei untersten Schichten des CAN Referenz-Models.

Die Konvertierung der Mess-System-Information in das CAN-Protokoll (CAN 2.0A) geschieht über einen CAN-Kontroller. Die Funktion des CAN-Kontrollers wird durch einen Watchdog überwacht.

Das CANopen Kommunikationsprofil (CiA Standard DS 301) basiert auf dem CAN Application Layer (CAL) und beschreibt, wie die Dienste von Geräten benutzt werden. Das CANopen Profil erlaubt die Definition von Geräteprofilen für eine dezentralisierte E/A.

Das Mess-System mit CANopen Protokoll unterstützt das Geräteprofil für Encoder (CiA Draft Standard 406, Version 3.2). **Die Mess-Systeme unterstützen zum Teil auch den erweiterten Funktionsumfang in Klasse C2.**

Die Kommunikations-Funktionalität und Objekte, welche im Encoderprofil benutzt werden, werden in einer EDS-Datei (Electronic Data Sheet) beschrieben. Wird ein CANopen Konfigurations-Hilfsprogramm benutzt (z.B. CANSETTER), kann der Benutzer die Objekte (SDO´s) des Mess-Systems auslesen und die Funktionalität programmieren.

6.2 EDS-Datei

Die EDS-Datei (elektronisches Datenblatt) enthält alle Informationen über die Mess-System-spezifischen Parameter sowie Betriebsarten des Mess-Systems. Die EDS-Datei wird durch das CANopen-Netzwerkkonfigurationswerkzeug eingebunden, um das Mess-System ordnungsgemäß konfigurieren bzw. in Betrieb nehmen zu können.

Die EDS-Datei hat den Dateinamen

- **"HLT1100-V01-R04.EDS"**

Die Dateien befinden sich auf der Software/Support CD: Mat.-Nr. 3505546

Die EDS-Datei (Dateiname **HLT1100-VXX-RYY.EDS**, XX = Version, YY = Release) inklusive der vorliegenden CANopen-Schnittstellenbeschreibung befindet sich zum Download auf unserer Homepage unter http://www.hydac.de/de-de/service/download-software-auf-anfrage/software/software-download/electronic.html.

7 Profilspezifische Prozessdaten-Objekte

Im Mess-System sind zwei Prozessdaten-Objekte (PDO) implementiert. Eine wird für die Asynchron-Übertragung und die andere für die Synchron-Übertragungsfunktionen benötigt.

Der Istwert wird im Binärcode übertragen:

COB-ID	Positionsausgabewert			
11 Bit	Byte 0	Byte 1	Byte 2	Byte 3
	2^7 bis 2^0	2^{15} bis 2^8	2^{23} bis 2^{16}	2^{31} bis 2^{24}

7.1 Erstes Sende-Prozessdaten-Objekt (asynchron)

Dieses PDO überträgt den Mess-System-Istwert asynchron. Der Timerwert ist im Index 1800h Sub 05h und im Index 6200h gespeichert.

Index	Subindex	Kommentar	Standardwert	Attr.
1800h	0	Anz. unterstützter Einträge	5	ro
	1	COB-ID benützt durch PDO 1	180h + Node-ID	ro
	2	Übertragungsart	254	ro
	3	Sperrzeit	0	rw
	5	Event timer	0	rw
1A00h	0	Anz. abgebildeter Objekte	1	ro
	1	Positionswert	60200120h	ro

7.2 Zweites Sende-Prozessdaten-Objekt (synchron)

Dieses PDO überträgt den Mess-System-Istwert synchron (auf Anforderung). Anforderung über Remote-Frame oder SYNC-Telegramm.

Index	Subindex	Kommentar	Standardwert	Attr.
1801h	0	Anz. unterstützter Einträge	5	ro
	1	COB-ID benützt durch PDO 2	280 + Node-ID	ro
	2	Übertragungsart	1	ro
	3	Sperrzeit	0	rw
	5	Event timer	0	rw
1A01h	0	Anz. abgebildeter Objekte	1	ro
	1	Positionswert	60200120h	ro

8 Kommunikationsspezifische Standard-Objekte (CiA DS-301)

Folgende Tabelle zeigt eine Übersicht der unterstützten Indexe im Kommunikationsprofilbereich:

Index (h)	Name	Seite
1000	Gerätetyp	43
1001	Fehlerregister	43
1003	Vordefiniertes Fehlerfeld	44
1005	COB-ID SYNC-Nachricht	44
1008	Hersteller Gerätenamen	45
1009	Hardwareversion	45
100A	Softwareversion	45
100C	Guard-Time (Überwachungszeit)	45
100D	Life-Time-Faktor (Zeitdauer-Faktor)	45
1010	Parameter abspeichern	46
1011	Parameter zurücksetzen	47
1018	Identity Objekt	48
1800	1. PDO Kommunikationsparameter	41
1801	2. PDO Kommunikationsparameter	41
1A00	1. PDO Mapping Parameter	41
1A01	2. PDO Mapping Parameter	41
1F80	Autostart	48

Tabelle 8: Kommunikationsspezifische Standard-Objekte

8.1 Objekt 1000h: Gerätetyp

Beinhaltet Information über den Gerätetyp. Das Objekt mit Index 1000h beschreibt den Gerätetyp und seine Funktionalität. Es besteht aus einem 16 Bit Feld, welches das benutzte Geräteprofil beschreibt (Geräteprofil-Nr. 406 = 196h) und ein zweites 16 Bit Feld, welches Informationen über den Gerätetyp liefert (Gerätetyp: Multi-Sensor Encoder Interface 10 = Ah).

Unsigned32

Gerätetyp			
Geräte-Profil-Nummer		Encoder-Typ	
Byte 0	Byte 1	Byte 2	Byte 3
0196h		000Ah	

8.2 Objekt 1001h: Fehlerregister

Dieses Objekt beinhaltet das Fehlerregister für das Gerät. Falls ein Alarm-Bit (Objekt 6503) gesetzt wird, wird im Fehlerregister das Bit 5 gesetzt.

Unsigned8

Bit	Bedeutung
0	generischer Fehler
1	0
2	0
3	0
4	Kommunikation
5	geräteprofilspezifisch
6	0
7	Sensor

8.3 Objekt 1003h: Vordefiniertes Fehlerfeld

Dieses Objekt speichert den zuletzt aufgetretenen Mess-System-Fehler und zeigt den Fehler über das Emergency-Objekt an. Jeder neue Fehler überschreibt einen zuvor gespeicherten Fehler in Subindex 1. Subindex 0 enthält die Anzahl der aufgetretenen Fehler. Die Bedeutung der Fehlercodes kann aus der Tabelle 11, Seite 61 entnommen werden.

Index	Subindex	Kommentar	Typ
1003h	0	Anzahl der Fehler	Unsigned8
	1	Standard Fehlerfeld	Unsigned32

Subindex 0: Der Eintrag in Subindex 0 beinhaltet die Anzahl der aufgetretenen Fehler und registriert sie in Subindex 1.

Subindex 1: Das Fehlerfeld setzt sich aus einem 16 Bit Fehlercode und einer 16 Bit Zusatz-Fehlerinformation zusammen.

Unsigned32

Standard Fehlerfeld			
Byte 0	Byte 1	Byte 2	Byte 3
Fehlercode		Zusatz-Fehlerinformation	

8.4 Objekt 1005h: COB-ID SYNC Nachricht

Dieses Objekt definiert die COB-ID des Synchronisierung-Objekts (SYNC). Es definiert weiterhin, ob das Gerät die SYNC-Nachricht verarbeitet, oder ob das Gerät die SYNC-Nachricht erzeugt. Das Mess-System unterstützt jedoch nur die Verarbeitung von SYNC-Nachrichten und verwendet den 11-Bit-Identifier.

Unsigned32
MSB LSB

31	30	29	28-11	10-0
1	0	0	0	00 1000 0000

Bit 31 = 1, Gerät verarbeitet die SYNC-Nachricht
Bit 30 = 0, Gerät erzeugt keine SYNC-Nachricht
Bit 29 = 0, 11 Bit ID (CAN 2.0A)
Bit 28 –11 = 0
Bit 10 – 0 = 11 Bit SYNC-COB-IDENTIFIER, Standardwert = 080h

Wenn ein SYNC-Telegramm mit der Identifier, definiert in diesem Objekt (080h), und Datenlänge = 0 vom Gerät empfangen worden ist, wird der Positionswert des Mess-Systems einmalig durch das zweite Sende-Prozessdaten-Objekt (Objekt 1801h) übertragen.

8.5 Objekt 1008h: Hersteller Gerätenamen

Enthält den Hersteller Gerätenamen (visible string),
Übertragung per „Segment Protokoll“.

8.6 Objekt 1009h: Hersteller Hardwareversion

Enthält die Hersteller Hardwareversion (visible string),
Übertragung per „Segment Protokoll“.

8.7 Objekt 100Ah: Hersteller Softwareversion

Enthält die Hersteller Softwareversion (visible string),
Übertragung per „Segment Protokoll“.

8.8 Objekt 100Ch: Guard-Time (Überwachungszeit)

Die Objekte der Indexe 100Ch und 100Dh beinhalten die Guard-Time in Milli-Sekunden und den Live-Time-Faktor (Zeitdauer-Faktor). Der Live-Time-Faktor multipliziert mit der Guard-Time ergibt die Zeitdauer für das Node-Guarding-Protokoll. Standardwert = 0.

Unsigned16

Guard-Time	
Byte 0	Byte 1
2^7 bis 2^0	2^{15} bis 2^8

8.9 Objekt 100Dh: Life-Time-Faktor (Zeitdauer-Faktor)

Der Live-Time-Faktor multipliziert mit der Guard-Time ergibt die Zeitdauer für das Node-Guarding-Protokoll. Standardwert = 0.

Unsigned8

Life-Time-Faktor
Byte 0
2^7 bis 2^0

8.10 Objekt 1010h: Parameter abspeichern

Dieses Objekt unterstützt das Abspeichern von Parametern in den nichtflüchtigen Speicher (EEPROM).

Index	Subindex	Kommentar	Typ
1010h	0	größter unterstützter Subindex	Unsigned8
	1	alle Parameter speichern	Unsigned32

Subindex0 (nur lesen)**:** Der Eintrag in Subindex 0 enthält den größten unterstützten Subindex. Wert =1.

Subindex1: Beinhaltet den Speicherbefehl.

Unsigned32

MSB LSB

Bits	31-2	1	0
Wert	= 0	0	1

Bei Lesezugriff liefert das Gerät Informationen über seine Speichermöglichkeit.

Bit 0 = 1, das Gerät speichert Parameter nur auf Kommando. Dies bedeutet, wenn Parameter durch den Benutzer geändert worden sind und das Kommando "Parameter abspeichern" nicht ausgeführt worden ist, nach dem nächsten Einschalten der Betriebsspannung, die Parameter wieder die alten Werte besitzen.



Bei Schreibzugriff speichert das Gerät die Parameter in den nichtflüchtigen Speicher. Dieser Vorgang dauert ca. 3s. In dieser Zeit ist das Mess-System auf dem Bus nicht ansprechbar.

Um eine versehentliche Speicherung der Parameter zu vermeiden, wird die Speicherung nur ausgeführt, wenn eine spezielle Signatur in das Objekt geschrieben wird. Die Signatur heißt "save".

Unsigned32

MSB LSB

e	v	a	s
65h	76h	61h	73h

Beim Empfang der richtigen Signatur speichert das Gerät die Parameter ab. Schlug die Speicherung fehl, antwortet das Gerät mit Abbruch der Übertragung, Fehlerklasse 6, Fehlerkennung 6 (Hardwarefehler), siehe auch Objekt 6503h.

Wurde eine falsche Signatur geschrieben, verweigert das Gerät die Speicherung und antwortet mit Abbruch der Übertragung, Fehlerklasse 8, Fehlerkennung 0.

8.11 Objekt 1011h: Parameter zurücksetzen

Mit diesem Objekt werden die Werte des Kommunikations-, Sensor- und Anwendungsprofils zurückgesetzt.

Index	Subindex	Kommentar	Typ
1011h	0	größter unterstützter Subindex	Unsigned8
	1	alle Parameter zurücksetzen	Unsigned32

Subindex0 (nur lesen)**:** Der Eintrag in Subindex 0 enthält den größten unterstützten Subindex. Wert = 1.

Subindex1: Beinhaltet den Ladebefehl.

Unsigned32

MSB LSB

Bits	31-2	1	0
Wert	= 0	0	1

Bei Lesezugriff liefert das Gerät Informationen über seine Rücksetzmöglichkeit.

Bit 0 = 1, das Gerät setzt Parameter nur auf Kommando zurück. Dies bedeutet, wenn Parameter durch den Benutzer geändert worden sind und das Kommando "Parameter rücksetzen" ausgeführt worden ist, werden die Parameter wieder die Werte der Werkseinstellung besitzen.



Bei Schreibzugriff speichert das Gerät die Parameter in den nichtflüchtigen Speicher. Dieser Vorgang dauert ca. 3s. In dieser Zeit ist das Mess-System auf dem Bus nicht ansprechbar.

Um ein versehentliches Rücksetzen der Parameter zu vermeiden, wird das Rücksetzen nur ausgeführt, wenn eine spezielle Signatur in das Objekt geschrieben wird. Die Signatur heißt "load".

Unsigned32

MSB LSB

d	a	o	l
64h	61h	6Fh	6Ch

Beim Empfang der richtigen Signatur werden die Parameter des Gerätes zurückgesetzt. Schlug das Rücksetzen fehl, antwortet das Gerät mit Abbruch der Übertragung, Fehlerklasse 6, Fehlerkennung 6 (Hardwarefehler), siehe auch Objekt 6503h.

Wurde eine falsche Signatur geschrieben, verweigert das Gerät das Rücksetzen und antwortet mit Abbruch der Übertragung, Fehlerklasse 8, Fehlerkennung 0.

8.12 Objekt 1018h: Identity Objekt

Dieses Objekt enthält generelle Informationen über das Gerät.

Index	Subindex	Kommentar	Typ
1018h	0	größter unterstützte Subindex	Unsigned8
	1	Vendor-ID	Unsigned32
	2	Produkt-Code	Unsigned32
	3	Revisionnummer	Unsigned32
	4	Seriennummer	Unsigned32

Subindex0: Der Eintrag in Subindex 0 enthält den größten unterstützten Subindex. Wert = 4.

Subindex1: Bei Lesezugriff liefert das Gerät die Vendor-ID des Herstellers. Die Vendor-ID von HYDAC-ELECTRONIC GmbH ist 218 bzw. 0xDA.

Subindex2: Bei Lesezugriff liefert das Gerät Informationen über den Produktcode Mit zugehörigem Geräteprofil.

Subindex3: Bei Lesezugriff liefert das Gerät Informationen über die Revisionsnummer.

Subindex4: Bei Lesezugriff liefert das Gerät Informationen über die Seriennummer.

8.13 Objekt 1F80: NMT-Startup (read / write)

Wird Bit 2 gesetzt, so wird automatisch bei Erreichen des „Pre-Operational“ Status in den „Operational“ Status gewechselt. Erlaubte Werte sind: 0x8 und 0xC

Index	Subindex	Kommentar	Typ
1F80h	0	NMT-Startup	Unsigned32

Bit 2 = 0 Automatisch in den Operational Mode wechseln

Bit 2 = 1 Nicht automatisch in den Operational Mode wechseln

9 Parametrierung und Konfiguration

Standardisierter Encoder-Profilbereich (CiA DS-406)

Die Einträge der Dateiliste von 6000h bis 65FFh werden von jedem Encoder genutzt. Die Einträge sind allgemein für Encoder.

Die untenstehende Übersicht zeigt alle vom HLT 1100 unterstützten Einträge:

Index (h)	Name	Seite
Parameter		
6000	Betriebsparameter	50
6005	Einstellung der Messschritte	50
6010	Presetwert	51
6020	Positionswert	52
6030	Geschwindigkeitswert	52
6200	Cyclic-Timer	53
Diagnose		
6500	Betriebsstatus	53
6501	Mess-Schritt	53
6503	Alarme	54
6504	Unterstützte Alarme	55
6505	Warnungen	55
6506	Unterstützte Warnungen	55
6507	Profil- und Softwareversion	56
6508	Betriebszeit	56
650A	Hersteller-Offsetwert	56
650B	Serien-Nummer	56
650C	Offsetwert	56

Tabelle 9: Encoder-Profilbereich

9.1 Objekt 6000h: Betriebsparameter

Die Betriebsparameter beinhalten die Funktion für die Zählrichtung.

Unsigned16

Bit	Funktion	Bit = 0	Bit = 1
0 - 2	reserviert		
3	Zählrichtung	steigend	fallend
4 - 15	reserviert		

Zählrichtung:
Die Zählrichtung definiert für Linear-Mess-Systeme, ob steigende oder fallende Positionswerte zum Ende des Stabes ausgegeben werden.

9.2 Objekt 6005h: Einstellung der Messschritte

Dieses Objekt enthält generelle Informationen über die eingestellten Schrittweiten für Position und Geschwindigkeit.

Index	Subindex	Kommentar	Typ
6005h	0	größter unterstützte Subindex	Unsigned8
	1	Positionsschrittweite	Unsigned32
	2	Geschwindigkeitsschrittweite	Unsigned32

Subindex0: Der Eintrag in Subindex 0 enthält den größten unterstützten Subindex. Wert = 2.

Subindex1: Der Wert für die Positionsschrittweite wird als Vielfaches von 0,001µm angegeben.

Subindex2: Der Wert für die Geschwindigkeitsschrittweite wird als Vielfaches von 0,01mm/s angegeben.

9.3 Objekt 6010h: Presetwert



WARNUNG !

Gefahr von Körperverletzung und Sachschaden durch einen Istwertsprung bei Ausführung der Preset-Justage-Funktion!

- Die Preset-Justage-Funktion sollte nur im Mess-System-Stillstand ausgeführt werden, bzw. muss der resultierende Istwertsprung programmtechnisch und anwendungstechnisch erlaubt sein!

Die Presetfunktion wird verwendet, um den Mess-System-Wert auf einen beliebigen Positionswert innerhalb des Bereiches von 0 bis Messlänge in Schritten — 1 zu setzen. Der Ausgabe-Positionswert wird auf den Parameter "Presetwert" gesetzt, wenn auf dieses Objekt geschrieben wird.

Signed32

Presetwert			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 bis 2^0	2^{15} bis 2^8	2^{23} bis 2^{16}	2^{31} bis 2^{24}

Index	Subindex	Kommentar	Typ
6010h	0	größter unterstützte Subindex	Usigned8
	1	Presetwert	Signed32

Subindex0: Der Eintrag in Subindex 0 enthält den größten unterstützten Subindex. Wert = 1.

Subindex1: Der Eintrag in Subindex 1 enthält den Presetwert für die Positionsausgabe.

9.4 Objekt 6020h: Positionswert

Das Objekt 6020h "Positionswert" definiert den Ausgabe-Positionswert für die Kommunikationsobjekte 1800h und 1801h.

Signed32

Positionswert			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 bis 2^0	2^{15} bis 2^8	2^{23} bis 2^{16}	2^{31} bis 2^{24}

Index	Subindex	Kommentar	Typ
6020h	0	größter unterstützte Subindex	Usigned8
	1	Positionswert	Signed32

Subindex0: Der Eintrag in Subindex 0 enthält den größten unterstützten Subindex. Wert = 1.

Subindex1: Der Eintrag in Subindex 1 enthält den Positionswert für die Positionsausgabe.

9.5 Objekt 6030h: Geschwindigkeitswert

Das Objekt 6030h "Geschwindigkeitswert" definiert den optionalen Geschwindigkeitsausgabewert für die Kommunikationsobjekte 1800h und 1801h.

Signed16

Geschwindigkeitswert	
Byte 0	Byte 1
2^7 bis 2^0	2^{15} bis 2^8

Index	Subindex	Kommentar	Typ
6030h	0	größter unterstützte Subindex	Usigned8
	1	Geschwindigkeitswert	Signed16

Subindex0: Der Eintrag in Subindex 0 enthält den größten unterstützten Subindex. Wert = 1.

Subindex1: Der Eintrag in Subindex 1 enthält den Geschwindigkeitswert für die Geschwindigkeitsausgabe.

9.6 Objekt 6200h: Cyclic-Timer

Definiert den Parameter "Cyclic-Timer". Eine asynchrone Übertragung des Positionswertes wird eingestellt, wenn der Cyclic-Timer auf > 0 programmiert wird. Es können Werte zwischen 1 ms und 65535 ms ausgewählt werden. Standardwert = 0.

z.B.: 1 ms = 1 h
256 ms = 100 h

Wenn das Mess-System mit dem Kommando NODE-START gestartet wird und der Wert des Cyclic-Timers > 0 ist, überträgt das erste Sende-Prozessdaten-Objekt (Objekt 1800h) die Mess-System-Position.

Dieses Objekt ist identisch mit dem Objekt 1800 Subindex 5.

9.7 Objekt 6500h: Betriebsstatus

Dieses Objekt enthält den Betriebsstatus des Mess-Systems und beinhaltet Informationen über die intern programmierten Parameter.

Unsigned16

Bit	Funktion	Bit = 0	Bit = 1
0 - 2	reserviert		
3	Zählrichtung	steigend	fallend
4 - 15	reserviert		

9.8 Objekt 6501h: Mess-Schritt, linear

Für Linear-Mess-Systeme zeigt das Objekt 6501h den Mess-Schritt an, welcher durch das Mess-System ausgegeben wird. Der Mess-Schritt wird in nm (0,001µm) angegeben. Beispiel: 1 µm = 00 00 03 E8 h

Unsigned32

Mess-Schritt			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 bis 2^0	2^{15} bis 2^8	2^{23} bis 2^{16}	2^{31} bis 2^{24}

Standardwert: 186A0 h = 100 000 = 0,1 mm

9.9 Objekt 6503h: Alarme

Das Objekt 6503h liefert zusätzlich zur „Emergency-Meldung“ weitere Alarm-Meldungen. Ein Alarm wird gesetzt, wenn eine Störung im Mess-System zum falschen Positionswert führen könnte. Falls ein Alarm auftritt, wird das zugehörige Bit solange auf logisch „High“ gesetzt, bis der Alarm gelöscht und das Mess-System bereit ist, einen richtigen Positionswert auszugeben.

Unsigned16

Bit	Funktion	Bit = 0	Bit = 1
0	Positionsfehler	Nein	Ja
1	Reserviert für weitere Verwendung		
2	Reserviert für weitere Verwendung		
3	Reserviert für weitere Verwendung		
4	Reserviert für weitere Verwendung		
5	Reserviert für weitere Verwendung		
6	Reserviert für weitere Verwendung		
7	Reserviert für weitere Verwendung		
8	Reserviert für weitere Verwendung		
9	Reserviert für weitere Verwendung		
10	Reserviert für weitere Verwendung		
11	Reserviert für weitere Verwendung		
12	Herstellerspezifische Funktionen		
13	Herstellerspezifische Funktionen		
14	Herstellerspezifische Funktionen		
15	Herstellerspezifische Funktionen		

Positionsfehler

Das Bit wird gesetzt, wenn beim Mess-System kein Magnet erkannt wird.

9.10 Objekt 6504h: Unterstützte Alarme

Das Objekt 6504h beinhaltet Informationen über die Alarme, die durch das Mess-System unterstützt werden.

Unsigned16

Bit	Funktion	Bit = 0	Bit = 1
0	Positionsfehler	Nein	Ja
1	Reserviert für weitere Verwendung		
2	Reserviert für weitere Verwendung		
3	Reserviert für weitere Verwendung		
4	Reserviert für weitere Verwendung		
5	Reserviert für weitere Verwendung		
6	Reserviert für weitere Verwendung		
7	Reserviert für weitere Verwendung		
8	Reserviert für weitere Verwendung		
9	Reserviert für weitere Verwendung		
10	Reserviert für weitere Verwendung		
11	Reserviert für weitere Verwendung		
12	Herstellerspezifische Funktionen		
13	Herstellerspezifische Funktionen		
14	Herstellerspezifische Funktionen		
15	Herstellerspezifische Funktionen		

9.11 Objekt 6505h: Warnungen

Dieses Objekt wird nicht unterstützt.
Bei Lesezugriff ist der Wert immer "0".

9.12 Objekt 6506h: Unterstützte Warnungen

Dieses Objekt wird nicht unterstützt.
Bei Lesezugriff ist der Wert immer "0".

9.13 Objekt 6507h: Profil- und Softwareversion

Dieses Objekt enthält in den ersten 16 Bits die implementierte Profilversion des Mess-Systems. Sie ist kombiniert mit einer Revisionsnummer und einem Index.

z.B.: Profilversion: 3.2
Binärcode: 0000 0011 0000 0010
Hexadezimal: 3 2

Die zweiten 16 Bits enthalten die implementierte Softwareversion des Mess-Systems. Nur die letzten 4 Ziffern sind verfügbar.

z.B.: Softwareversion: 01.04
Binärcode: 0000 0001 0000 0010
Hexadezimal: 01 04

Die komplette Softwareversion ist in Objekt 100Ah enthalten, siehe Seite 45.

Unsigned32

Profilversion		Softwareversion	
Byte 0	Byte 1	Byte 2	Byte 3
2^7 bis 2^0	2^{15} bis 2^8	2^7 bis 2^0	2^{15} bis 2^8

9.14 Objekt 6508h: Betriebszeit

Die Betriebszeit-Funktion wird nicht verwendet, der Betriebszeitwert wird auf den Maximalwert gesetzt (FF FF FF FF h).

9.15 Objekt 650Ah: Modul-Identifizierung

Die Sub-Indices 1 – 3 enthalten Informationen über herstellerspezifische Einstellungen.

9.16 Objekt 650Bh: Serien-Nummer

Bei Lesezugriff liefert das Gerät Informationen über die Seriennummer.
Entspricht Objekt 1018 Subindex 4.

9.17 Objekt 650Ch: Offsetwert

Dieses Objekt enthält den Offsetwert, der durch die Preset-Funktion berechnet wird. Der Offsetwert wird gespeichert und kann vom Mess-System gelesen werden.

Index	Subindex	Kommentar	Typ
650Ch	0	größter unterstützte Subindex	Usigned8
	1	Offsetwert	Signed32

10 Emergency-Meldung

Emergency-Meldungen werden beim Auftreten einer geräteinternen Störung ausgelöst und werden von dem betreffenden Anwendungsgerät an die anderen Geräte mit höchster Priorität übertragen.

Emergency-Meldung								
Byte	0	1	2	3	4	5	6	7
Inhalt	Emergency-Fehlercode Objekt 1003h, Byte 0-1		Fehler-Register Objekt 1001h	0	0	0	0	0

COB-Identifier = 080h + Node-ID

Wenn das Mess-System einen internen Fehler erkennt, wird eine Emergency-Meldung mit dem Fehlercode des Objekts 1003h (Vordefiniertes Fehlerfeld) und dem Fehler-Register (Objekt 1001h) übertragen. Zusätzlich zum Emergency-Objekt wird im Alarm-Objekt 6503H das zugehörige Bit gesetzt.

Wenn der Fehler nicht mehr vorhanden ist, überträgt das Mess-System eine Emergency-Meldung mit dem Fehlercode "0" (Reset Fehler / kein Fehler) und Fehler-Register "0".

11 Übertragung des Mess-System-Positionswertes

Bevor die Mess-System-Position übertragen werden kann, muss das Mess-System mit dem „Node-Start“-Kommando gestartet werden.

Node-Start Protokoll

COB-Identifier = 0	
Byte 0	Byte 1
1	Node-ID

Das Node-Start Kommando mit der Node-ID des Mess-Systems (Slave) startet nur dieses Gerät.

Das Node-Start Kommando mit der **Node-ID = 0** startet alle Slaves die am Bus angeschlossen sind.

Nach dem Node-Start Kommando überträgt das Mess-System den Positionswert einmal mit der COB-ID des Objekts 1800h.

Jetzt kann der Positionswert auf verschiedene Arten übertragen werden:

1. Asynchron-Übertragung

Das erste Sende-Prozessdaten-Objekt (Objekt 1800h) überträgt den Positionswert des Mess-Systems in Zeitintervallen. Der Timerwert wird definiert durch den Wert des Cyclic-Timers (Objekt 6200h oder 1800/5). Diese Übertragung startet automatisch nach dem Kommando Node-Start und der Wert des Cyclic-Timers ist > 0.

Der Standardwert der COB-ID ist 180h + Node-ID.

Objekt	Funktions-Code	COB-ID	Index Kommunikations-Parameter
PDO1 (tx)	0011bin	181h – 1FFh	1800h

Um die Übertragung der Mess-System-Position kurzzeitig zu stoppen, kann die Ausgabe durch Timerwert = 0 im Objekt 6200h (oder 1800/5) unterbrochen werden.

2. Synchron-Übertragung

Das zweite Sende-Prozessdaten-Objekt (Objekt 1801h) überträgt einmalig den Positionswert des Mess-Systems nach einer Anforderung (Remote / Sync):

- Das Mess-System empfängt ein Remote-Frame mit der COB-ID (Standardwert 280h + Node-ID).

Objekt	Funktions-Code	COB-ID	Index Kommunikations-Parameter
PDO2 (tx)	0101bin	281h – 2FFh	1801h

- Das Mess-System empfängt ein SYNC-Telegramm mit der COB-ID (Standardwert 080h), definiert in Objekt 1005h. Alle Slaves mit dieser SYNC-COB-ID übertragen den Positionswert.

Objekt	Funktions-Code	COB-ID	Index Kommunikations-Parameter
SYNC	0001bin	80h	1005

12 Fehlerursachen und Abhilfen

12.1 SDO-Fehlercodes

Im Fall eines Fehlers (SDO Response CCD = 0x80) enthält der Datenbereich einen 4-Byte-Fehlercode. Folgende Fehler-Codes werden vom Mess-System unterstützt:

Fehlercode	Bedeutung	Abhilfe
0x0600 0006	EEPROM-Fehler	Mess-System-Spannung eventuell ausschalten, danach wieder einschalten. Wenn der Fehler trotz dieser Maßnahme wiederholt auftritt, muss das Mess-System getauscht werden.
0x0601 0000	Nicht unterstützter Zugriff auf ein Objekt	Überprüfen, welches Attribut für das entsprechende Objekt gültig ist: - rw: Lese- und Schreibzugriff - wo: nur Schreibzugriff - ro: nur Lesezugriff - Const: nur Lesezugriff .
0x0609 0011	Subindex nicht vorhanden	Überprüfen, welche Subindexe das entsprechende Objekt unterstützt.
0x0800 0000	Allgemeiner Fehler	Falsche Signatur beim Abspeichern der Parameter geschrieben, siehe Objekt 1010h: Parameter abspeichern, Seite 46.

Tabelle 10: SDO-Fehlercodes

12.2 Emergency-Fehlercodes

Emergency-Meldungen werden beim Auftreten einer geräteinternen Störung ausgelöst, Übertragungsformat siehe Kapitel „Emergency-Meldung“, Seite 57. Die Fehleranzeige wird über die Objekte

- Fehlerregister 0x1001, siehe Seite 43 und
- Vordefiniertes Fehlerfeld 0x1003, siehe Seite 44

vorgenommen.

12.2.1 Objekt 1001h: Fehlerregister

Das Fehlerregister zeigt bitkodiert den Fehlerzustand des Mess-Systems an. Es können auch mehrere Fehler gleichzeitig durch ein gesetztes Bit angezeigt werden. Der Fehlercode des zuletzt aufgetretenen Fehlers wird in Objekt 0x1003, Subindex 1 hinterlegt, die Anzahl der Fehler im Subindex 0. Im Moment des Auftretens wird ein Fehler durch eine EMCY-Nachricht signalisiert. Durch Lesen des Objekts 1001h wird der zuletzt gespeicherte Fehler in Objekt 0x1003, Subindex 0 gelöscht. Jede weitere Leseanforderung löscht einen weiteren Fehler aus der Liste. Mit Löschen des letzten Fehlers wird das Fehlerregister zurückgesetzt und eine EMCY-Nachricht mit Fehlercode „0x000“ übertragen.

Unsigned8

Bit	Bedeutung
0	generischer Fehler
1	0
2	0
3	0
4	Kommunikation
5	geräteprofilspezifisch
6	0
7	Sensor

12.2.2 Objekt 1003h: Vordefiniertes Fehlerfeld, Bits 0 – 15

Über das Emergency-Objekt wird immer nur der zuletzt aufgetretene Fehler angezeigt. Für jede EMCY-Nachricht die gelöscht wurde, wird eine Emergency-Meldung mit Fehlercode „0x0000“ übertragen. Das Ergebnis kann dem Objekt 0x1003 entnommen werden. Wenn kein Fehler mehr vorliegt, zeigt auch das Fehlerregister keinen Fehler mehr an.

Die Fehlerliste in Objekt 0x1003 kann auf verschiedene Arten gelöscht werden:

1. Schreiben des Wertes „0“ auf Subindex 0 im Objekt 0x1003
2. Ausführen des NMT-Dienstes „Reset Communication“, Kommando 0x82
3. Durch Lesen des Objekts 0x1001, nach dem der letzte Fehler gelöscht wurde

Fehlercode	Bedeutung	Abhilfe
0x0000	Fehler rückgesetzt / kein Fehler	-
0x8100	Kommunikationsfehler, die vom CAN-Controller ausgelöst werden.	- Knoten zurücksetzen mit Kommando 0x81, danach Knoten neu starten mit Kommando 0x01 - Mess-System-Spannung ausschalten, danach wieder einschalten.

Tabelle 11: Emergency-Fehlercodes

12.3 Alarm-Meldungen

Über das Objekt 6503h werden zusätzlich zur Emergency-Meldung weitere Alarm-Meldungen ausgegeben. Das entsprechende Fehlerbit wird gelöscht, wenn der Fehler nicht mehr anliegt.

Fehler	Ursache	Abhilfe
Bit 0 = 1, Positionsfehler	Ausfall von Abtastelementen im Mess-System. Mess-System hat keinen Magnet erkannt.	Magnet in den Messbereich bringen.
Bit 12 = 1, EEPROM-Fehler	Speicherbereich im internen EEPROM defekt	Versorgungsspannung eventuell ausschalten, danach wieder einschalten. Wenn der Fehler trotz dieser Maßnahme wiederholt auftritt, muss das Mess-System getauscht werden.
Bit 13 = 1 Parameterfehler	Programmierter Parameter außerhalb Bereich.	Min- bzw. Max.-Werte der einzelnen Parameter überprüfen.

12.4 Sonstige Störungen

Störung	Ursache	Abhilfe
Positionssprünge des Mess-Systems	starke Vibrationen	Vibrationen, Schläge und Stöße z.B. an Pressen, werden mit so genannten „Schockmodulen“ gedämpft. Wenn der Fehler trotz dieser Maßnahmen wiederholt auftritt, muss das Mess-System getauscht werden.
	elektrische Störungen EMV	Gegen elektrische Störungen helfen eventuell isolierende Flansche und Kupplungen aus Kunststoff, sowie Kabel mit paarweise verdrillten Adern für Daten und Versorgung. Die Schirmung und die Leitungsführung müssen nach den Aufbaurichtlinien für das jeweilige Feldbus-System ausgeführt sein.
	übermäßige axiale und radiale Belastung der Welle oder einen Defekt der Abtastung.	Kupplungen vermeiden mechanische Belastungen der Welle. Wenn der Fehler trotz dieser Maßnahme weiterhin auftritt, muss das Mess-System getauscht werden.

HYDAC ELECTRONIC GMBH
Hauptstr. 27
D-66128 Saarbrücken
Germany

Web: www.hydac.com
E-Mail: electronic@hydac.com
Tel.: +49 (0)6897 509-01
Fax.: +49 (0)6897 509-1726

HYDAC Service
Für Fragen zu Reparaturen steht Ihnen der HYDAC Service zur Verfügung.

HYDAC SERVICE GMBH
Hauptstr. 27
D-66128 Saarbrücken
Germany

Tel.: +49 (0)6897 509-1936
Fax.: +49 (0)6897 509-1933

Anmerkung

Die Angaben in dieser Bedienungsanleitung beziehen sich auf die beschriebenen Betriebsbedingungen und Einsatzfälle. Bei abweichenden Einsatzfällen und/oder Betriebsbedingungen wenden Sie sich bitte an die entsprechende Fachabteilung.

Bei technischen Fragen, Hinweisen oder Störungen nehmen Sie bitte Kontakt mit Ihrer HYDAC-Vertretung auf.

Technische Änderungen sind vorbehalten.



User Manual

(Translation of original instruction)



Linear Transducer Serie HLT 1100
with CANopen interface

- **Additional safety instructions**
- **Installation**
- **Commissioning**
- **Configuration / Parameterization**
- **Cause of faults and remedies**

- **Software/Support CD: Mat. No. 3505546**

Mat. No. 669824 Stand: 07.10.2010

Preface

Keep the manual in the vicinity of the instrument for immediate reference.

Please note: the specifications outlined in this documentation for the instrument technology are correct at the time of publishing. Deviations in technical specifications, illustrations and dimensions are therefore possible.

Should you find any errors whilst using this manual, or have any suggestions for improvements, please contact:

HYDAC ELECTRONIC GMBH
Technische Dokumentation
Hauptstraße 27
66128 Saarbrücken
-Deutschland-
Tel: +49(0)6897 / 509-01
Fax: +49(0)6897 / 509-1726
Email: electronic@hydac.com

Copyright protection

Subject to amendments

Any technical changes that serve the purpose of technical progress, reserved.

Font styles

Italic or **bold** font styles are used for the title of a document or are used for highlighting.

`Courier` font displays text, which is visible on the display or screen and software menu selections.

" < > " indicates keys on your computer keyboard (such as <RETURN>).

Trademarks

CANopen is a registered trademark of CAN in Automation e.V.

Contents

1 General information

The User Manual includes the following topics:

- Safety instructions in additional to the basic safety instructions defined in the Assembly Instructions
- Electrical characteristics
- Installation
- Commissioning
- Configuration / parameterization
- Causes of faults and remedies

As the documentation is arranged in a modular structure, this User Manual is supplementary to other documentation, such as product datasheets, dimensional drawings, leaflets and the assembly instructions etc.

The User Manual may be included in the customer's specific delivery package or it may be requested separately.

1.1 Applicability

This User Manual applies exclusively to the following measuring system models with ***CANopen*** interface:

- HLT 1100

The products are labeled with affixed nameplates and are components of a system.

The following documentation therefore also applies:

- the operator's operating instructions specific to the system,
- this User Manual,
- and the assembly instructions Mat. No. **669822**, which is enclosed when the device is delivered

1.2 References

1.	ISO 11898: Road Vehicles Interchange of Digital Information - Controller Area Network (CAN) for high-speed Communication, November 1993	
2.	Robert Bosch GmbH, CAN Specification 2.0 Part A and B, September 1991	
3.	CiA DS-201	V1.1, CAN in the OSI Reference Model, February 1996
4.	CiA DS-202-1	V1.1, CMS Service Specification, February 1996
5.	CiA DS-202-2	V1.1, CMS Protocol Specification, February 1996
6.	CiA DS-202-3	V1.1, CMS Encoding Rules, February 1996
7.	CiA DS-203-1	V1.1, NMT Service Specification, February 1996
8.	CiA DS-203-2	V1.1, NMT Protocol Specification, February 1996
9.	CiA DS-204-1	V1.1, DBT Service Specification, February 1996
10.	CiA DS-204-2	V1.1, DBT Protocol Specification, February 1996
11.	CiA DS-205-1	V1.1, LMT Service Specification, February 1996
12.	CiA DS-205-2	V1.1, LMT Protocol Specification, February 1996
13.	CiA DS-206	V1.1, Recommended Layer Naming Conventions, February 1996
14.	CiA DS-207	V1.1, Application Layer Naming Conventions, February 1996
15.	CiA DS-301	V3.0, CANopen Communication Profile based on CAL, October 1996
16.	CiA DS-406	V2.0, CANopen Profile for Encoder, May 1998



1.3 Abbreviations and definitions

HLT 1100	Linear-Absolute Measuring System, for integration in cylinders
EC	***E***uropean ***C***ommunity
EMC	***E***lectro ***M***agnetic ***C***ompatibility
ESD	***E***lectro ***S***tatic ***D***ischarge
IEC	***I***nternational ***E***lectrotechnical ***C***ommission
VDE	German Electrotechnicians Association

CAN specific

CAL	CAN Application Layer. The application layer for CAN-based networks as specified by CiA in Draft Standard 201 ... 207.
CAN	Controller Area Network. Data link layer protocol for serial communication as specified in ISO 11898.
CiA	CAN in Automation international manufacturer and user organization e.V.: non-profit association for Controller Area Network (CAN).
CMS	CAN-based Message Specification. One of the service elements of the application layer in the CAN Reference Model.
COB	Communication Object. (CAN Message) A unit of transportation in a CAN Network. Data must be sent across a Network inside a COB.
COB-ID	COB-Identifier. Identifies a COB uniquely in a Network. The identifier determines the priority of that COB in the MAC sub-layer too.
DBT	Distributor. One of the service elements of the application in the CAN Reference Model. It is the responsibility of the DBT to distribute COB-ID´s to the COB´s that are used by CMS.
EDS	***E***lectronic-***D***ata-***S***heet
LSS	Layer Setting Services. One of the service elements of the application in the CAN Reference Model. It serves to configure parameters of each layer in the CAN Reference Model.
NMT	Network Management. One of the service elements of the application in the CAN Reference Model. It performs initialization, configuration and error handling in a CAN network.
PDO	Process Data Object. Object for data exchange between several devices.
SDO	Service Data Object. Peer to peer communication with access to the Object Dictionary of a device.



2 Additional safety instructions

2.1 Definition of symbols and instructions



WARNING!
means that death, serious injury or major damage to property could occur if the stated precautions are not met.



CAUTION !
means that minor injuries or damage to property can occur if the stated precautions are not met.



Note
indicates important information's or features and application tips for the product used.

2.2 Additional instructions for proper use

The measurement system is designed for operation with CANopen networks according to the International Standard ISO/DIS 11898 and 11519-1 up to max. 1 Mbit/s. The profile corresponds to the ***"CANopen Device Profile for Encoder CiA DS-406 V2.0A"***.

The technical guidelines for the structure of the CANopen network from the CAN User Organization CiA are always to be observed in order to ensure safe operation.



Proper use also includes:

- observing all instructions in this User Manual,
- observing the assembly instructions. The ***"Basic safety instructions"*** in particular must be read and understood prior to commencing work.

2.3 Organizational measures

- This User Manual must always kept accessible at the site of operation of the measurement system.
- Prior to commencing work, personnel working with the measurement system must have read and understood
  - the assembly instructions, in particular the chapter ***"Basic safety instructions"***,
  - and this User Manual, in particular the chapter "Additional safety instructions".

This particularly applies for personnel who are only deployed occasionally, e.g. at the parameterization of the measurement system.

3 Technical data

Supply voltage: .... 12 / 24 V DC

Current consumption without load: . < 100 mA

Measuring principle:....................... magnetostrictive

Resolution:...................................... 0.1 mm

Output data Position – 32 Bit Integer
Speed – 16 Bit Integer

CANopen: .... EN 50325-4
Bus connection: .... ISO 11898-1, ISO 11898-2
CAN Specification 2.0 A:............. 11-Bit Identifier
Communication-Profile: CiA DS 301 V4.2
Device Profile .... CiA DS 406 V3.2
LSS: CiA DS 305 V2.2
Node-ID: .... 1…127
Baud rate: .... 20, 50, 125, 250, 500, 800, 1000 kbit/s

Connection: Copper cable 4 x 0.5 mm²

Special features: .... Programming of the following parameters via the CAN-BUS:
- Node adress
- Baudrate
- Code sequence
- Preset value
- PDO mapping

EMC: .... EN 61326/ISO 14982/ISO 7637/ISO 11452/EN 50121



4 CANopen information

CANopen was developed by the CiA and is standardized since at the end of 2002 in the European standard EN 50325-4.

As communication method CANopen uses the layers 1 and 2 of the CAN standard which was developed originally for the use in road vehicles (ISO 11898-2). In the automation technology these are extended by the recommendations of the CiA industry association with regard to the pin assignment and transmission rates.
In the area of the application layer CiA has developed the standard CAL (CAN Application Layer).

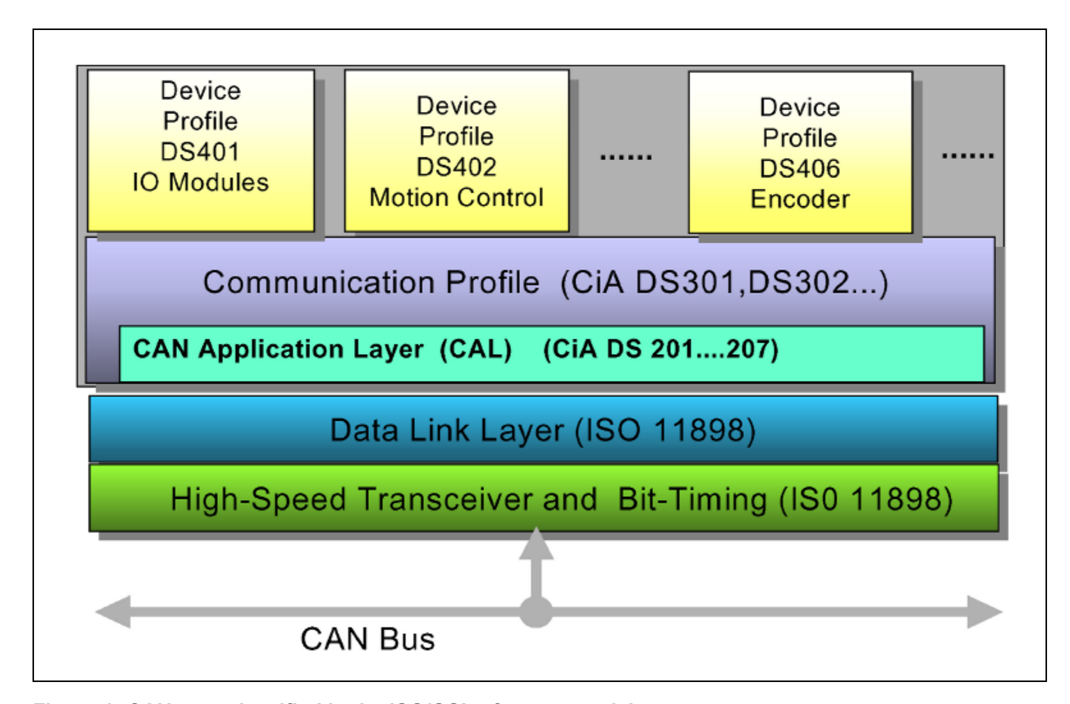


Figure 1: CANopen classified in the ISO/OSI reference model

In case of CANopen at first the communication profile as well as a "Build instructions" for device profiles was developed, in which with the structure of the object dictionary and the general coding rules the common denominator of all device profiles is defined.



4.1 CANopen – Communication profile

The CANopen communication profile (defined in CiA DS-301) regulates the devices data exchange. Here real time data (e.g. position value) and parameter data (e.g. code sequence) will be differentiated. To the data types, which are different from the character, CANopen assigns respectively suitable communication elements.

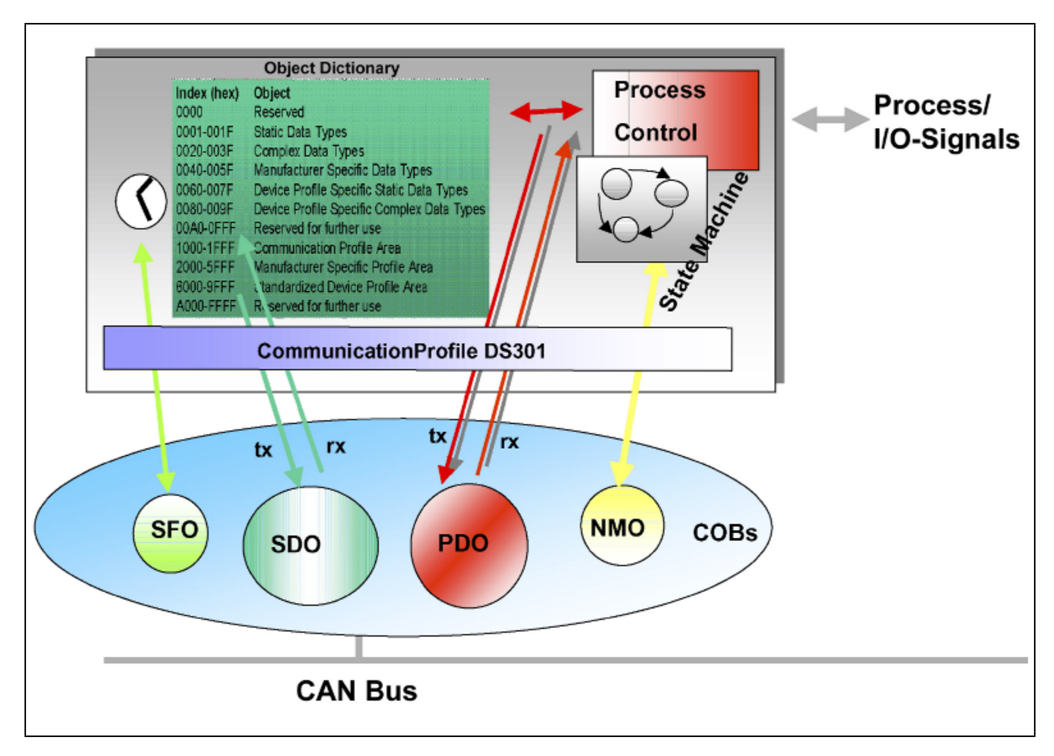


Figure 2: Communication profile

Special **F**unction **O**bject **(SFO)**

- Synchronization (SYNC)
- Emergency (EMCY) Protocol

Network **M**anagement **O**bject **(NMO)**

e.g.

- Life / Node-Guarding
- Boot-Up,…
- Error Control Protocol

4.2 Process- and Service-Data-Objects

Process-Data-Object (PDO)

Process-Data-Objects manage the process data exchange, such as the cyclical transmission of the position value.
The process data exchange with the CANopen PDOs is "CAN pure", therefore without protocol overhead. All broadcast characteristics of CAN remain unchanged. A message can be received and evaluated by all devices at the same time.
From the measuring system the two transmitting process data objects 1800h for asynchronous (event-driven) position transmission and 1801h for the synchronous (upon request) position transmission are used.

Service-Data-Object (SDO)

Service-Data-Objects manage the parameter data exchange, such as the non-cyclical execution of the Preset function.
For parameter data of arbitrary size with the SDO an efficient communication mechanism is available. For this between the configuration master and the connected devices a service data channel for the parameter communication is available. The device parameters can be written with only one telegram handshake into the object dictionary of the devices or can be read out from this.

Important characteristics of the SDO and PDO

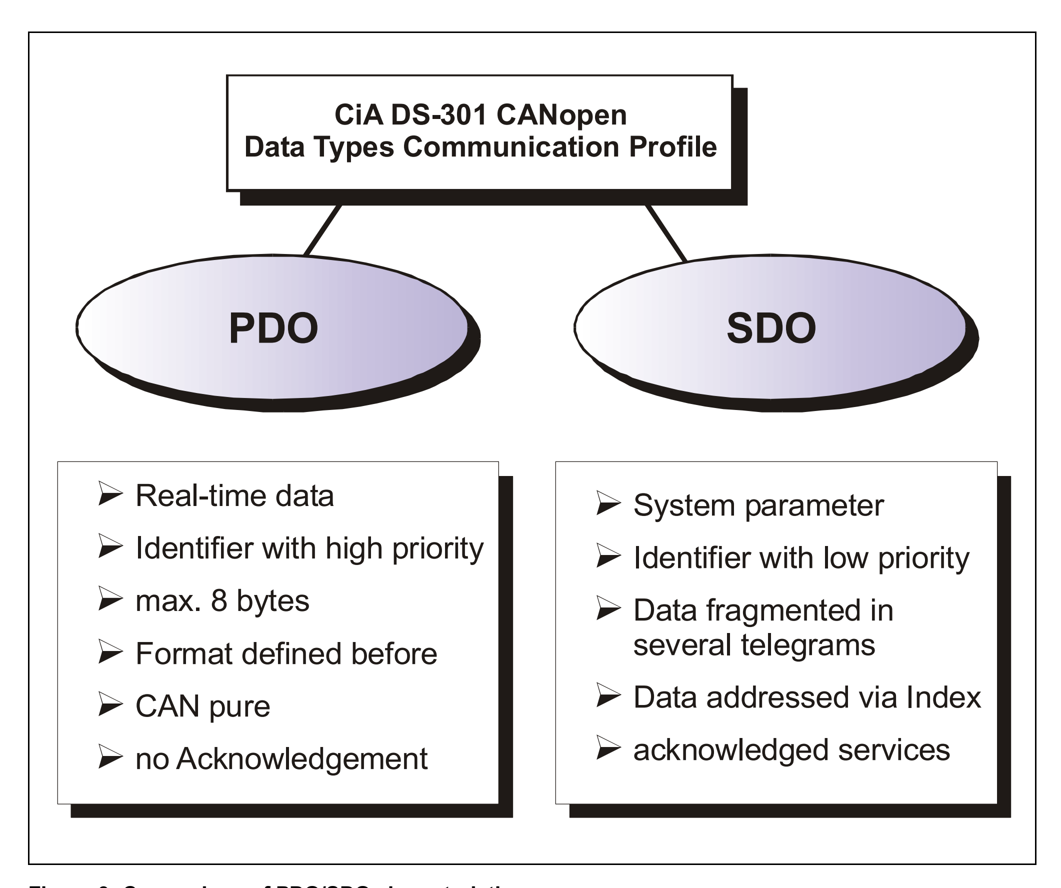


Figure 3: Comparison of PDO/SDO characteristics

4.3 Object Dictionary

The object dictionary structures the data of a CANopen device in a clear tabular arrangement. It contains all device parameters as well as all current process data, which are accessible thereby also about the SDO.

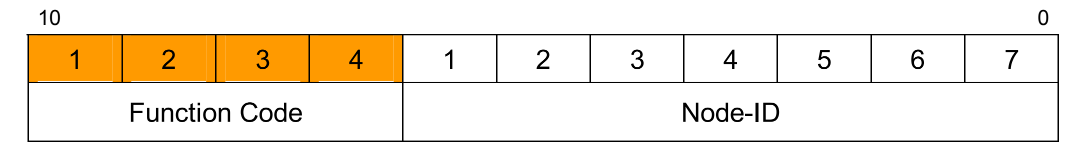
Index	Object	
0000_h	not used	Common to all devices
0001_h - $025F_h$	Data type definitions	
0260_h - $0FFF_h$	Reserved	
1000_h - $1FFF_h$	Communication profile area	
2000_h - $5FFF_h$	Manufacturer specific profile area	Device specific
6000_h - $9FFF_h$	Standardized device profile area	
$A000_h$ - $BFFF_h$	Standardized interface profile area	
$C000_h$ - $FFFF_h$	Reserved	

Figure 4: Structure of the Object Dictionary

4.4 CANopen default identifier

CANopen devices can be used without configuration in a CANopen network. Just the setting of a bus address and the baud rate is required. From this node address the identifier allocation for the communication channels is derived.

COB-Identifier = Function Code + Node-ID



Examples

Object	Function Code	COB-ID	Index Communication Parameter
NMT	0000bin	0	–
SYNC	0001bin	80h	1005
PDO1 (tx)	0011bin	181h – 1FFh	1800h

4.5 Transmission of SDO messages

The transmission of SDO messages is done by the CMS "Multiplexed Domain" protocol (CIA DS202-2).
With SDOs objects from the object dictionary can be read or written. It is an acknowledged service. The so-called ***SDO client*** specifies in its request the parameter, the access method (read/write) and if necessary the value. The so-called ***SDO server*** performs the write or read access and answers the request with a response. In the error case an error code gives information about the cause of error. Transmit-SDO and Receive-SDO are distinguished by their function codes.

The measuring system (slave) corresponds to the SDO server and uses the following function codes:

Function codes	COB-ID	Meaning
11 (1011 bin)	**0x580 + Node ID**	**Slave → SDO Client**
12 (1100 bin)	**0x600 + Node ID**	**SDO Client → Slave**

Table 1: COB-IDs for Service Data Object (SDO)

4.5.1 SDO message format

The data field with max. 8 byte length of a CAN message is used by a SDO as follows:

CCD	Index		Sub-Index	Data			
Byte 0	Byte 1 Low	Byte 2 High	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

Table 2: SDO message

The ***command code*** (***CCD***) identifies whether the SDO is to be read or written. In addition with a writing order, the number of bytes which can be written is encoded in the CCD.
At the SDO response the CCD reports whether the request was successful. In the case of a reading order the CCD gives additionally information about the number of bytes, which could be read:

CCD	Meaning	Valid for
0x23	Write 4 bytes	SDO Request
0x2B	Write 2 bytes	SDO Request
0x2F	Write 1 byte	SDO Request
0x60	Writing successfully	SDO Response
0x80	Error	SDO Response
0x40	Reading request	SDO Request
0x43	4 byte data read	SDO response upon reading request
0x4B	2 byte data read	SDO response upon reading request
0x4F	1 byte data read	SDO response upon reading request

Table 3: SDO command codes

In the case of an error (SDO response CCD = 0x80) the data field contains a 4-byte error code, which gives information about the error cause. Meaning of the error codes see table Table 10 on page 60.

Segment Protocol, Data segmentation

Some objects contain data which are larger than 4 bytes. To be able to read these data, the "Segment Protocol" must be used.
As a usual SDO service, at first the read operation is started with the command code = 0x40. About the response the number of data segments and the number of bytes to be read is reported. With following reading requests the individual data segments can be read. A data segment consists respectively of 7 bytes.

Example of reading a data segment:

Telegram 1

CCD	Meaning	Valid for
0x40	Reading request, initiation	SDO Request
0x41	1 data segment available The number of bytes which can be read is indicated in the bytes 4 to 7.	SDO Response

Telegram 2

CCD	Meaning	Valid for
0x60	Reading request	SDO Request
0x01	No further data segment available. The bytes 1 to 7 contain the requested data.	SDO Response

4.5.2 Read SDO

Initiate Domain Upload

Request Protocol format:

COB-Identifier = 600h + Node-ID

Read SDO´s								
Byte	0	1	2	3	4	5	6	7
Contents	Code	Index		Sub-Index	Data 0	Data 1	Data 2	Data 3
	40h	Low	High	Byte	0	0	0	0

The Read SDO telegram has to be send to the slave.

The slave answers with the following telegram:

Response Protocol format:

COB-Identifier = 580h + Node-ID

Read SDO´s								
Byte	0	1	2	3	4	5	6	7
Contents	Code	Index		Sub-Index	Data 0	Data 1	Data 2	Data 3
	4xh	Low	High	Byte	Data	Data	Data	Data

Format Byte 0:

MSB LSB

7	6	5	4	3	2	1	0
0	1	0	0	n		1	1

n = number of data bytes (bytes 4-7) that does not contain data

If only 1 data byte (Data 0) contains data the value of byte 0 is "4FH".

If byte 0 = 80h the transfer has been aborted.

4.5.3 Write SDO

Initiate Domain Download

Request Protocol format:

COB-Identifier = 600h + Node-ID

Write SDO´s								
Byte	0	1	2	3	4	5	6	7
Contents	Code	Index		Sub-Index	Data 0	Data 1	Data 2	Data 3
	2xh	Low	High	Byte	0	0	0	0

Format Byte 0:

MSB LSB

7	6	5	4	3	2	1	0
0	0	1	0	n		1	1

n = number of data bytes (bytes 4-7) that does not contain data.

If only 1 data byte (Data 0) contains data the value of byte 0 is "2FH".

The Write SDO telegram has to be send to the slave.

The slave answers with the following telegram:

Response Protocol format:

COB-Identifier = 580h + Node-ID

Read SDO´s								
Byte	0	1	2	3	4	5	6	7
Contents	Code	Index		Sub-Index	Data 0	Data 1	Data 2	Data 3
	60h	Low	High	Byte	0	0	0	0

If byte 0 = 80h the transfer has been aborted.



4.6 Network management, NMT

The network management supports a simplified Boot-Up of the net. With only one telegram for example all devices can be switched into the Operational condition.

After Power on the measuring system is first in the "Pre-Operational" condition (2).

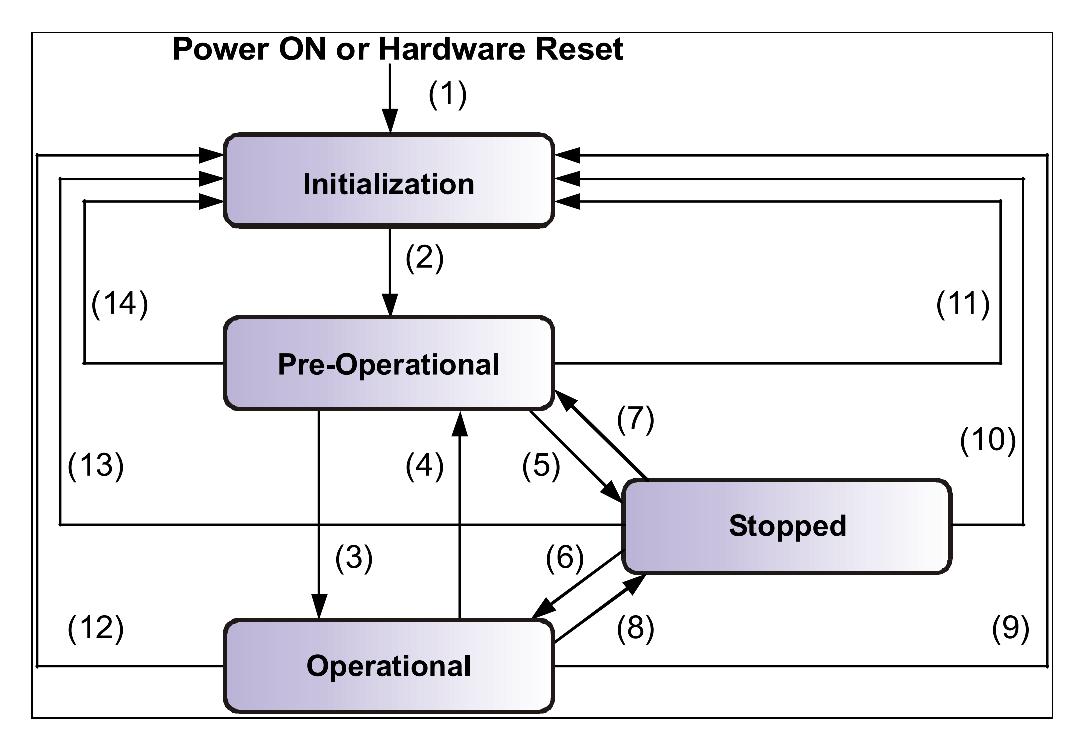


Figure 5: Boot-Up mechanism of the network management

State	Description
(1)	At Power on the initialization state is entered autonomously
(2)	Initialization finished - enter PRE-OPERATIONAL automatically
(3),(6)	Start_Remote_Node --> Operational
(4),(7)	Enter_PRE-OPERATIONAL_State --> Pre-Operational
(5),(8)	Stop_Remote_Node
(9),(10),(11)	Reset_Node
(12),(13),(14)	Reset_Communication

4.6.1 Network management services

The ***network management*** (***NMT***) has the function to initialize, start, stop and monitor nodes of a CANopen network.

NMT services are initiated by a ***NMT master***, which identifies individual nodes (***NMT slave***) about their Node-ID. A NMT message with the Node ID 0 refers to **all** NMT slaves.

The measuring system corresponds to a NMT slave.

4.6.1.1 NMT device control services

The NMT services for device control use the **COB-ID 0** and get thus the highest priority.

By the data field of the CAN message only the first two bytes are used:

CCD	Node ID
Byte 0	Byte 1

The following commands are defined:

CCD	Meaning	State
-	At Power on the initialization state is entered autonomously	(1)
-	Initialization finished - enter PRE-OPERATIONAL automatically	(2)
0x01	**Start Remote Node** Node is switched into the OPERATIONAL state and the normal net-operation is started.	(3),(6)
0x02	**Stop Remote Node** Node is switched into the STOPPED state and the communication is stopped. An active connecting monitoring remains active.	(5),(8)
0x80	**Enter PRE-OPERATIONAL** Node is switched into the PRE-OPERATIONAL state. All messages can be used, but no PDOs.	(4),(7)
0x81	**Reset Node** Set values of the profile parameters of the object on default values. Afterwards transition into the RESET COMMUNICATION state.	(9),(10), (11)
0x82	**Reset Communication** Node is switched into the RESET COMMUNICATION state. Afterwards transition into the INITIALIZATION state, first state after Power on.	(12),(13), (14)

Table 4: NMT device control services



4.6.1.2 NMT Node / Life guarding services

With the Node/Life guarding a NMT master can detect the failure of a NMT slave and/or a NMT slave can detect the failure of a NMT master:

- **Node Guarding and Life Guarding:**
  With these services a NMT master monitors a NMT slave

At the **Node Guarding** the NMT master requests the state of a NMT slave in regular intervals. The toggle bit 2^7 in the “Node Guarding Protocol” toggles after each request:

Example:
0x85, 0x05, 0x85 … --> no error
0x85, 0x05, 0x05 … --> error

Additionally if the **Life Guarding** is active, the NMT slave requests the state of a NMT master in regular intervals, otherwise the slave changes into the PRE-OPERATIONAL state.

The NMT services for Node/Live guarding use the function code `1110 bin:` **COB-ID 0x700+Node ID**.

Index	Description	
0x100C	**Guard Time [ms]**	At termination of the time interval **Life Time = Guard Time x Life Time Factor [ms]** the NMT slave expects a state request by the master. Guard Time = 0: No monitoring active Life Time = 0: Life guarding disabled
0x100D	**Life Time Factor**	

Table 5: Parameter for NMT services

4.7 Layer setting services (LSS) and protocols

The LSS -services and protocols, documented in CIA DS-305 V2.2, are used to inquire or to change the settings of certain parameters of the local layers of a LSS slave by a LSS master via the CAN network.

Following parameters are supported:

- Node-ID
- Baud rate
- LSS -address, under the terms of identity object 1018h

Access to the LSS slave is made thereby by its LSS address, consisting of:

- Vendor-Id
- Product-Name
- Revision Number
- Serial-Number

The measuring system supports the following services:

Switch mode services

- Switch mode selective
  - To response a specific of LSS-Slave
- Switch mode global
  - To response all of LSS-Slaves

Configuration services

- Configure Node-ID
- Configure bit timing parameters
- Activate bit timing parameters
- Store configured parameters

Inquiry services

- Inquire LSS-address
- Inquire Node-ID

Identification services

- LSS identify remote slave
  - Identification of LSS-Slaves within an certain array
- LSS identify slave
  - Response of all LSS-Slaves to the previous command
- LSS identify non-configured remote slave
  - Identification of non-configured LSS-Slaves, Node-ID = FFh
- LSS identify non-configured slave
  - Response of all LSS-Slaves to the previous command

4.7.1 LSS Modes and Services

The LSS is equivalent to a state machine and defines the behavior of a LSS-Slave. The state behavior is controlled by LSS COBs produced by the LSS Master, or NMT Cobs generated by a NMT Master, or local NMT state transitions.

The LSS Modes support the following states:

(0) Initial: Pseudo-State , shows the activation of the FSAs
(1) LSS waiting: Support for all services as indicated below
(2) LSS configuration: Support for all services as indicated below
(3) Final: Pseudo-State, , shows the deactivation of the FSAs

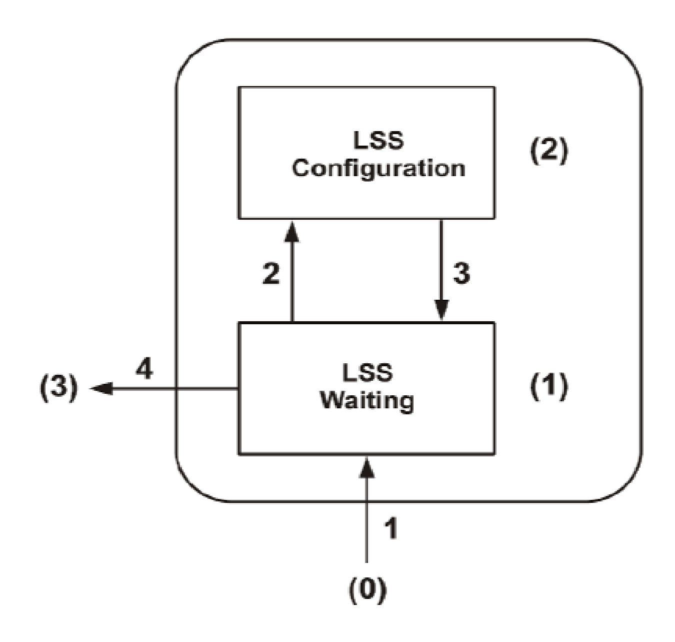


Figure 6: LSS Modes

State behavior of the supported services

Services	Waiting	Configuration
Switch mode global	Yes	Yes
Switch mode selective	Yes	No
Activate bit timing parameters	No	Yes
Configure bit timing parameters	No	Yes
Configure Node-Id	No	Yes
Store configuration	No	Yes
Inquire LSS-address	No	Yes
LSS identify remote slave	Yes	Yes
LSS identify slave	Yes	Yes
LSS identify non-configuration remote slave	Yes	Yes
LSS identify non-configured slave	Yes	Yes

LSS FSA state transitions

Transition	Events	Actions
1	Automatic transition after the initialization on entering either the state NMT PRE OPERATIONAL state or NMT STOPPED state, or NMT RESET COMMUNICATION state with Node-ID = FFh.	no
2	LSS 'switch state global' command with parameters 'configuration_switch' or 'switch state selective' command	no
3	LSS 'switch state global' command with parameter 'waiting_switch'	no
4	Automatic transition when an invalid Node-ID has been changed and the new Node-ID could be successfully stored in nonvolatile memory AND the state of LSS waiting has been requested.	no

Once the LSS FSA further state transitions of the NMT FSA on NMT PRE into OPERATIONAL STOPPED and experienced vice versa, the result is not the re-entry into the LSS FSA.

4.7.2 Transmission of LSS services

By means of LSS services, the LSS master requests services to be performed by the LSS slave. Communication between LSS master and LSS slave is made by means of implemented LSS protocols.
Similar as in the case of SDO transmitting, also here two COB-IDs for sending and receiving are used:

COB-ID	Meaning
0x7E4	**LSS-Slave → LSS-Master**
0x7E5	**LSS-Master → LSS-Slave**

Table 6: COB-IDs for LSS services

4.7.2.1 LSS message format

The data field with max. 8 byte length of a CAN message is used by a LSS service as follows:

CS	Data						
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

Table 7: LSS message

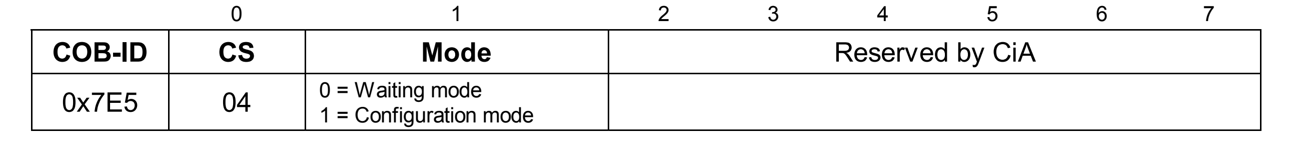
Byte 0 contains the ***Command-Specifier*** (CS), afterwards 7 byte data are following.

4.7.3 Switch mode protocols

4.7.3.1 Switch mode global protocol

The given protocol has implemented the *Switch mode global service* and controls the state behavior of the LSS slave. By means of the LSS master all LSS slaves in the network can be switched into *Waiting Mode* or *Configuration Mode*.

LSS-Master --> LSS-Slave



	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Mode**	Reserved by CiA					
0x7E5	04	0 = Waiting mode 1 = Configuration mode						

4.7.3.2 Switch mode selective protocol

The given protocol has implemented the *Switch mode selective service* and controls the state behavior of the LSS slave. By means of the LSS master only this LSS slave in the network can be switched into *Configuration Mode*, whose LSS address attributes equals the LSS address.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Vendor-Id**				Reserved by CiA		
0x7E5	64	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Product-Code**				Reserved by CiA		
0x7E5	65	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number**				Reserved by CiA		
0x7E5	66	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number**				Reserved by CiA		
0x7E5	67	LSB			MSB			

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	68							

Error Code
- 0: Protocol successfully completed
- 1…254: reserved
- 255: application specific error occurred

Specific Error
if Error Code = 255 --> application specific error occurred,
otherwise reserved by CiA

4.7.4 Configuration protocols

4.7.4.1 Configure Node-Id -address protocol

The given protocol has implemented the *Configure NMT-address service*. By means of the LSS master the Node-ID of a single LSS slave in the network can be configured. Only one device is to be switched into *Configuration Mode*. For storage of the new Node-ID the *Store configuration protocol* must be transmitted to the LSS slave.

LSS Master --> LSS Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Node-ID**	Reserved by CiA					
0x7E5	17	1…127 and 255						

LSS-Slave --> LSS Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Error Code**	**Spec. Error**	Reserved by CiA				
0x7E4	17							

Error Code
- 0: Protocol successfully completed
- 1…254: reserved
- 255: application specific error occurred

Specific Error
if Error Code = 255 --> application specific error occurred,
otherwise reserved by CiA

4.7.4.2 Configure bit timing parameters protocol

The given protocol has implemented the *Configure bit timing parameters service*. By means of the LSS master the Baud rate of a single LSS slave or of all LSS-Slaves in the network can be configured. For storage of the new Baud rate the *Store configuration protocol* must be transmitted to the LSS slave.

LSS-Master --> LSS Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Table Selector**	**Table Index**	Reserved by CiA				
0x7E5	19	0	0…7					

LSS Slave --> LSS Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Error Code**	**Spec. Error**	Reserved by CiA				
0x7E4	19							

Table Selector
- 0: Standard CiA Baud rate table

Table Index
- 0: 1 Mbit/s
- 1: 800 kbit/s
- 2: 500 kbit/s
- 3: 250 kbit/s
- 4: 125 kbit/s
- 5: reserved
- 6: 50 kbit/s
- 7: 20 kbit/s

Error Code
- 0: Protocol successfully completed
- 1: selected Baud rate not supported
- 2…254: reserved
- 255: application specific error occurred

Specific Error
if Error Code = 255 --> application specific error occurred, otherwise reserved by CiA

4.7.4.3 Activate bit timing parameters protocol

The given protocol has implemented the `Activate bit timing parameters service`. The protocol activates the Baud rate which was configured about the `Configure bit timing parameters protocol` and is performed with all LSS slaves in the network which are in the `Configuration Mode`.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Switch Delay [ms]**		Reserved by CiA				
0x7E5	21	LSB	MSB					

Switch Delay

The parameter `Switch Delay` defines the length of two delay periods (D1, D2) with equal length. These are necessary to avoid operating the bus with differing Baud rate parameters.
After the time D1 and an individual processing duration, the switching internally in the LSS slave is performed. After the time D2 the LSS slave responses with CAN-messages and the new configured Baud rate.

It is necessary:
Switch Delay > longest occurring processing duration of a LSS slave

4.7.4.4 Store configuration protocol

The given protocol has implemented the `Store configuration service`. By means of the LSS master the configured parameters of a single LSS slave in the network can be stored into the non-volatile memory. Only one device is to be switched into `Configuration Mode`.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	23							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Error Code**	**Spec. Error**	Reserved by CiA				
0x7E4	23							

Error Code

0: Protocol successfully completed
1: `Store configuration` not supported
2…254: reserved
255: application specific error occurred

Specific Error

if Error Code = 255 --> application specific error occurred,
otherwise reserved by CiA

4.7.5 Inquire LSS address protocols

4.7.5.1 Inquire Identity Vendor-ID protocol

The given protocol has implemented the `Inquire LSS address service`. By means of the LSS master the Identity Vendor-ID of a single LSS slave in the network can be read-out. Only one device is to be switched into `Configuration Mode`.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	90							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Vendor-Id** (= Index 1018h:01)				Reserved by CiA		
0x7E4	90	LSB			MSB			

4.7.5.2 Inquire Identity Product-Code protocol

The given protocol has implemented the `Inquire LSS address service`. By means of the LSS master the Product-Name of a single LSS slave in the network can be read-out. Only one device is to be switched into `Configuration Mode`.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	91							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Product-Code** (= Index 1018h:02)				Reserved by CiA		
0x7E4	91	LSB			MSB			

4.7.5.3 Inquire Identity Revision-Number protocol

The given protocol has implemented the *Inquire LSS address service*. By means of the LSS master the Identity Revision-Number of a single LSS slave in the network can be read-out. Only one device is to be switched into *Configuration Mode*.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	92							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number** (= Index 1018h:03)				Reserved by CiA		
0x7E4	92	LSB			MSB			

4.7.5.4 Inquire Identity Serial-Number protocol

The given protocol has implemented the *Inquire LSS address service*. By means of the LSS master the Serial-Number of a single LSS slave in the network can be read-out. Only one device is to be switched into *Configuration Mode*.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	93							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number** (= Index 1018h:04)				Reserved by CiA		
0x7E4	93	LSB			MSB			

4.7.5.5 Inquire Node-ID protocol

The given protocol has implemented the `Inquire Node-ID.` The specified protocol is implemented to Inquire Node-ID service. By means of the LSS-Master the Node-ID of a single LSS-Slave can readout in the network. This may only find one LSS Slave in `configuration mode`.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E5	94							

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Node-ID**	Reserved by CiA					
0x7E4	94	1…127 und 255	MSB					

Node-ID
Corresponds to the Node-ID of the selected device. If the Node-ID just has been changed about *`the Configure Node-ID service`*, the original Node-ID is returned. Only after implementation of the NMT-Service Reset Communication (0x82), the current Node-ID is returned.

4.7.6 Identification protocols

4.7.6.1 LSS identify remote slave protocol

The given protocol has implemented the *LSS identify remote slaves service*. By means of the LSS master LSS slaves in the network can be identified within a certain range. All LSS slaves with matching Vendor-ID, Product-Code, Revision-No. and Serial-No. Range, response with the *LSS identify slave protocol*.

LSS-Master --> LSS-Slave

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Vendor-Id**				Reserved by CiA		
0x7E5	70	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Product-Code**				Reserved by CiA		
0x7E5	71	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number-Low**				Reserved by CiA		
0x7E5	72	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Revision-Number-High**				Reserved by CiA		
0x7E5	73	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number-Low**				Reserved by CiA		
0x7E5	74	LSB			MSB			

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**Serial-Number-High**				Reserved by CiA		
0x7E5	75	LSB			MSB			

4.7.6.2 LSS identify slave protocol

The given protocol has implemented the *LSS identify slave service*. All LSS slaves with matching LSS attributes given in the *LSS identify remote slaves protocol*, response with this protocol.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	79							

4.7.6.3 LSS identify non-configured remote slave protocol

The specified protocol has implemented the *LSS identify non-configured remote slave service*. By means of the LSS-Master all unconfigured LSS-Slaves (Node-ID = identified FFh) in the network are identified. The related LSS-Slaves answer with the *LSS identify non-configured remote slave protocol*

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	76							

4.7.6.4 LSS identify non-configured slave Protocol

The specified protocol has implemented the *LSS identify non-configured slave service.* All LSS-Slaves with an invalid Node-ID (FFh), answer with the protocol, after implementation the *LSS identify non-configured slave protocol*.

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	Reserved by CiA						
0x7E4	80							

4.7.6.5 LSS fastscan protocol

The specified protocol has implemented the *LSS fastscan service.* All LSS-Slaves, after identification by all 4 LSS subentries, change into *Configuration Mode.*

LSS-Slave --> LSS-Master

	0	1	2	3	4	5	6	7
COB-ID	**CS**	**ID Nummer**				**Bit Check**	**LSS Sub**	**LSS Next**
0x7E4	81	LSB			MSB			

4.8 Device profile

The CANopen device profiles describe the "what" of the communication. In the profiles the meaning of the transmitted data is unequivocal and manufacturer independently defined. So the basic functions of each device class

e.g. for encoder: ***CiA DS-406***

can be responded uniformly. On the basis of these standardized profiles CANopen devices can be accessed in an identical way over the bus. Therefore devices which support the same device profile are exchangeable with each other.

You can obtain further information on CANopen from the ***CAN in Automation*** User- and Manufacturer Association:

CAN in Automation
Am Weichselgarten 26
DE-91058 Erlangen

Tel. +49-9131-69086-0
Fax +49-9131-69086-79

Website: www.can-cia.org
e-mail: headquarters@can-cia.org

---

5 Installation / Preparation for start-up

The CANopen system is wired in bus topology with terminating resistors (120 ohms) at the beginning and at the end of the bus line. If it is possible, drop lines should be avoided. The cable is to be implemented as shielded twisted pair cable and should have an impedance of 120 ohms and a resistance of 70 mΩ/m. The data transmission is carried out about the signals CAN-H and CAN-L with a common GND as data reference potential. Optionally also the 24 V supply voltage can be carried.

In a CANopen network max. 127 slaves can be connected. The measuring system supports the Node-ID range from 1-127. The transmission rate can be adjusted via LSS-Protocol and supports the baud rates 20 kbit/s, 125 kbit/s, 500 kbit/s and 1 Mbit/s.

The length of a CANopen network is depending on the transmission rate and is represented in the following:

Cable cross section	20 kbit/s	125 kbit/s	500 kbit/s	1 Mbit/s
0.25 mm^2 – 0.34 mm^2	2500 m	500 m	100 m	25 m

The

- *ISO 11898,*
- *the recommendations of the CiA DR 303-1*
  (CANopen cabling and connector pin assignment)
- *and other applicable standards and guidelines are to be observed to insure safe and stable operation!*

In particular, the applicable EMC directive and the shielding and grounding guidelines must be observed!

5.1 Connection

The connection can be made with device specific pin assignment which is enclosed when the device is delivered.

For the supply shielded cables with twisted core pairs have to be used !

5.2 Switching on the supply voltage

After the connection and all settings have been carried out, the supply voltage can be switched on.

After power on and finishing the initialization, the measuring system goes into the PRE-OPERATIONAL state. This status is acknowledged by the Boot-Up message “**COB-ID 0x700+Node ID**”. If the measuring system detects an internal error, an emergency message with the error code will be transmitted (see chapter "Emergency Message", page 57).

In the PRE-OPERATIONAL state first only a parameter setting about Service-Data-Objects is possible. But it is possible to configure PDOs with the help of SDOs. If the measuring system was transferred into the OPERATIONAL state, also a transmission of PDOs is possible.

5.3 Setting the Node-ID and Baud rate by means of LSS services

5.3.1 Configuration of the Node-ID, sequence

Assumption:

- LSS address unknown
- only one LSS slave should be in the network
- the Node-ID 12 dec. shall be adjusted

Procedure:

- Perform service 04 `Switch mode global protocol`,
  Mode = 1, to switch the LSS slave into `Configuration Mode`.
- Perform service 17 `Configure Node-ID protocol`, Node-ID = 12.
  --> Wait for acknowledgement and check successfully execution,
  --> Error Code = 0.
- Perform service 23 `Store configuration protocol`.
  --> Wait for acknowledgement and check successfully execution,
  --> Error Code = 0.
- Switch the supply voltage of the LSS slave OFF and then ON again. Now the new configuration is activated.

5.3.2 Configuration of the Baud rate, sequence

Assumption:

- LSS address unknown
- only one LSS slave should be in the network
- the Baud rate 125 kbit/s shall be adjusted

Procedure:

- Perform service 04 *Switch mode global protocol*,
  Mode = 1, to switch the LSS slave into *Configuration Mode*.
- Perform service 19 *Configure bit timing parameters protocol*,
  Table Selector = 0, Table Index = 4
  --> Wait for acknowledgement and check successfully execution,
  --> Error Code = 0.
- Perform service 21 *Activate bit timing parameters protocol*, to switch the new Baud rate active.
- Perform service 23 *Store configuration protocol*.
  --> Wait for acknowledgement and check successfully execution,
  --> Error Code = 0.
- Switch the supply voltage of the LSS slave OFF and then ON again. Now the new configuration is activated.

6 Commissioning

6.1 CAN – interface

The CAN-Bus-Interface is defined by the international norm ISO/DIS 11898 and specifies the two lowest layers of the ISO/DIS CAN Reference Model.

The conversion of the measuring system information to the CAN message format (CAN 2.0A) is done by a CAN-controller. The function of the CAN-controller is controlled by a watchdog.

The CANopen Communication Profile (CIA standard DS 301) is a subset of CAN Application Layer (CAL) and describes, how the services are used by devices. The CANopen Profile allows the definition of device profiles for decentralized I/O.

The measuring system with CANopen-protocol support the Device Profile for Encoder (CIA Draft Standard Proposal 406, Version 3.2). **The measuring systems partially support the extended functions in Class C2 .**

The communication functionality and objects, which are used in the encoder profile, are described in a EDS-File (Electronic Data Sheet).
When using a CANopen Configuration Tool (e.g.: CANSETTER), the user can read the objects of the measuring system (SDOs) and program the functionality.

6.2 EDS file

The EDS (electronic datasheet) contains all information on the measuring system-specific parameters and the measuring system's operating modes. The EDS file is integrated using the CANopen network configuration tool to correctly configure or operate the measuring system.

The EDS file has the file name

- **"HLT1100-V01-R04.EDS"**

The file is on the Software/Support CD: Mat.-No. 3505546

The EDS- file (file name **HLT1100-VXX-RYY.EDS** XX = Version, YY = Release) description of CANopen-interface included can be downloaded on HYDAC homepage under http://www.hydac.com/de-en/support/download-software-on-demand/software/software-download/electronic.html.

7 Profile specific process data objects

Two process data objects (PDO) are implemented in the device. One is used for asynchronous transmission and the other one for the cyclic transmission functions.

The output position value is transmitted in binary code:

COB-ID	Output Position Value			
11 Bit	Byte 0	Byte 1	Byte 2	Byte 3
	2^7 to 2^0	2^{15} to 2^8	2^{23} to 2^{16}	2^{31} to 2^{24}

7.1 1st transmit Process-Data-Object (asynchronous)

This PDO transmit the position value of the measuring system in an asynchronous way. The cyclic timer is stored in Index 1800h Sub 05h and in index 6200h.

Index	Sub-Index	Comment	Default value	Attr.
1800h	0	number of supported entries	5	ro
	1	COB-ID used by PDO 1	180h + Node-ID	ro
	2	transmission type	254	ro
	3	inhibit time	0	rw
	5	Event timer	0	rw
1A00h	0	number of mapped objects	1	ro
	1	Position value	60200120h	ro

7.2 2nd transmit Process-Data-Object (cyclic)

This PDO transmit the position value of the measuring system in a cyclic way (on request). Request by remote frame and/or sync telegrams.

Index	Sub-Index	Comment	Default value	Attr.
1801h	0	number of supported entries	5	ro
	1	COB-ID used by PDO 2	280 + Node-ID	ro
	2	transmission type	1	ro
	3	inhibit time	0	rw
	5	Event timer	0	rw
1A01h	0	number of mapped objects	1	ro
	1	Position value	60200120h	ro

8 Communication specific standard objects (CiA DS-301)

Following table gives an overview on the supported indices in the Communication Profile Area:

Index (h)	Name	Page
1000	Device type	43
1001	Error register	43
1003	Pre-defined error field	44
1005	COB-ID SYNC message	44
1008	Device name	45
1009	Hardware version	45
100A	Software version	45
100C	Guard time	45
100D	Life time factor	45
1010	Store parameters	46
1011	Restore parameters	47
1018	Identity object	48
1800	1. PDO communication parameters	41
1801	2. PDO communication parameters	41
1A00	1. PDO Mapping parameters	41
1A01	2. PDO Mapping parameters	41
1F80	Autostart	48

Table 8: Communication specific standard objects

8.1 Object 1000h: Device type

Contains information about the device type. The object at index 1000h describes the type of device and its functionality. It is composed of a 16 bit field which describes the device profile that is used (Device Profile Number 406 = 196h) and a second 16 bit field which gives information on the type of encoder (encoder type: Multi-Sensor Encoder Interface 10 = Ah).

Unsigned32

Device Type			
Device Profile Number		Encoder Type	
Byte 0	Byte 1	Byte 2	Byte 3
196h		000Ah	

8.2 Object 1001h: Error register

This object contains the error register for the device. If an alarm bit is set (object 6503), bit 5 is set in the error register.

Unsigned8

Bit	Meaning
0	generic error
1	0
2	0
3	0
4	communication
5	device profile specific
6	0
7	sensor

8.3 Object 1003h: Pre-defined error field

This object saves the measuring system error occurred last and displays the error via the Emergency object. Each new error overwrites an error which was stored before in sub-index 1. Sub-index 0 contains the number of the occurred errors. Meaning of the error codes see Table 11, page 61.

Index	Sub-Index	Comment	Type
1003h	0	number of errors	Unsigned8
	1	standard error field	Unsigned32

Sub-index 0: The entry at sub-index 0 contains the number of errors that have occurred and recorded in sub-index 1.

Sub-index 1: The error are composed of a 16 bit error code and a 16 bit additional error information.

Unsigned32

Standard Error Field			
Byte 0	Byte 1	Byte 2	Byte 3
Error code		Additional Information	

8.4 Object 1005h: COB-ID SYNC message

This object defines the COB-ID of the Synchronization Object (SYNC). Further, it defines whether the device consumes the SYNC or whether the device generates the SYNC. However, the measuring system supports only the processing of SYNC-messages and uses the 11-bit identifier.

Unsigned32
MSB LSB

31	30	29	28-11	10-0
1	0	0	0	00 1000 0000

Bit 31 = 1, Device processes the SYNC message
Bit 30 = 0, Device does not generate the SYNC message
Bit 29 = 0, 11-bit ID (CAN 2.0A)
Bit 28 –11 = 0
Bit 10 – 0 = 11-bit SYNC-COB-IDENTIFIER, default value = 080H

If a SYNC-telegram with the identifier, defined in this object (080H), and data length = 0 has been received by the device, the position value of the measuring system is transmitted by the 2nd Transmit PDO (object 1801h), non-recurrent triggering.

8.5 Object 1008h: Device name

Contains the manufacturer device name (visible string),
transmission via “Segment Protocol”.

8.6 Object 1009h: Hardware version

Contains the manufacturer hardware version (visible string),
transmission via “Segment Protocol”.

8.7 Object 100Ah: Software version

Contains the manufacturer software version (visible string),
transmission via “Segment Protocol”.

8.8 Object 100Ch: Guard time

The objects at index 100CH and 100DH include the guard time in milli-seconds and the life time factor. The life time factor multiplied with the guard time gives the live time for the Node Guarding Protocol. Default value = 0.

Unsigned16

Guard Time	
Byte 0	Byte 1
2^7 to 2^0	2^{15} to 2^8

8.9 Object 100Dh: Life time factor

The life time factor multiplied with the guard time gives the life time for the node guarding protocol. Default value = 0.

Unsigned8

Life Time Factor
Byte 0
2^7 to 2^0

8.10 Object 1010h: Store parameters

This object supports the saving of parameters in non volatile memory (EEPROM).

Index	Sub-Index	Comment	Type
1010h	0	largest supported Sub-Index	Unsigned8
	1	save all parameters	Unsigned32

Sub-Index0 (only read): The entry at sub-index 0 contains the largest Sub-Index that is supported. Value = 1.

Sub-Index1: Contains the save command.

Unsigned32

MSB LSB

Bits	31-2	1	0
Value	= 0	0	1

By read access the device provides information about its saving capability.

Bit 0 = 1, the device saves parameters only on command. That means, if parameters have been changed by the user and no "Store Parameter Command" had been executed, at the next power on, the parameters will have there old values.



In case of write access the device stores the parameters to the non volatile memory. This procedure takes approx. 3s. In this time the measuring system isn't accessible at the bus.

In order to avoid storage of parameters by mistake, storage is only executed when a specific signature is written to the object. The signature is "save".

Unsigned32

MSB LSB

e	v	a	s
65h	76h	61h	73h

On reception of the correct signature, the device stores the parameters. If the storing failed, the device responds with abort domain transfer, error class 6, error code 6 (hardware fault). See also object 6503h.

If a wrong signature is written, the device refuses to store and responds with abort domain transfer, error class 8, error code 0.

8.11 Object 1011h: Restore parameters

This object restore the values of the communication, sensor and application profile.

Index	Sub-Index	Comment	Type
1011h	0	largest supported Sub-Index	Unsigned8
	1	restore all parameters	Unsigned32

Sub-Index0 (only read): The entry at sub-index 0 contains the largest Sub-Index that is supported. Value = 1.

Sub-Index1: Contains the restore command.

Unsigned32

MSB LSB

Bits	31-2	1	0
Value	= 0	0	1

By read access the device provides information about its restore capability.

Bit 0 = 1, the device restores parameters only on command. That means, if parameters have been changed by the user and "Restore Parameter Command" had been executed, at the next power on, the parameters will have the values of factory setting.



In case of write access the device stores the parameters to the non volatile memory. This procedure takes approx. 3s. In this time the measuring system isn't accessible at the bus.

In order to avoid restore of parameters by mistake, restore is only executed when a specific signature is written to the object. The signature is "load".

Unsigned32

MSB LSB

d	a	o	l
64h	61h	6Fh	6Ch

On reception of the correct signature, the device restores the parameters. If the restoring failed, the device responds with abort domain transfer, error class 6, error code 6 (hardware fault). See also object 6503h.

If a wrong signature is written, the device refuses to restore and responds with abort domain transfer, error class 8, error code 0.

8.12 Object 1018h: Identity Object

This object provides general identification information of the CANopen device.

Index	Sub-Index	Comment	Type
1018h	0	highest sub-index supported	Unsigned8
	1	Vendor-ID	Unsigned32
	2	Product-Code	Unsigned32
	3	Order-No.	Unsigned32
	4	Serial-No.	Unsigned32

Sub-index0: The entry at sub-index 0 contains the largest Sub-Index that is supported: Value = 4.

Sub-index1: Contains the Vendor-ID of the manufacturer. The Vendor-ID for HYDAC-ELECTRONIC GmbH is 218 = 0xDA.

Sub-index2: Provides information about the product code with the associated device profile.

Sub-index3: Provides information about the order number.

Sub-index4: Provides information about the serial number.

8.13 Objekt 1F80: NMT-Startup (read / write)

Will be set bit 2, automatically by reaching the “Pre-operational” state, will be change into “Operational” state. Permitted values are: 0x8 and 0xC.

Index	Subindex	Comment	Type
1F80h	0	NMT-Startup	Unsigned32

Bit 2 = 0 Automatically change into operational mode

Bit 2 = 1 No automatically change into operational mode

9 Parameterization and configuration

Standardized encoder profile area (CiA DS-406)

Each encoder shares the dictionary entries from 6000h to 65FFh. These entries are common to encoders.

The overview of all entries supported by HLT 1100 is shown below:

Index (h)	Name	Page
Parameters		
6000	Operating parameters	50
6005	Measuring step settings	50
6010	Preset value	51
6020	Position value	52
6030	Speed value	52
6200	Cyclic timer	53
Diagnostics		
6500	Operating state	53
6501	Measuring step	53
6503	Alarms	54
6504	Supported alarms	55
6505	Warnings	55
6506	Supported warnings	55
6507	Profile and software version	56
6508	Operating time	56
650A	Modul identification	56
650B	Serial number	56
650C	Offset value	56

Table 9: Encoder profile area

9.1 Object 6000h: Operating parameters

The Operating Parameters contain the function for code sequence.

Unsigned16

Bit	Function	Bit = 0	Bit = 1
0 -2	Reserved		
3	Code sequence	increasing	decreasing
4- 15	Reserved		

Code sequence:
For linear measuring systems the code sequence defines whether increasing or decreasing position values to the end of the rod are output.

9.2 Object 6005h: Measuring step setting

This object contains generel information about the setting up of increment for position and speed.

Index	Sub index	Comment	Type
6005h	0	Highest sub-index supported	Unsigned8
	1	Position increment	Unsigned32
	2	Speed increment	Unsigned32

Sub index0: The entry at sub-index 0 contains the highest sub-index that is supported. Value = 2.

Sub index1: The value of position increment is given as a multiple of 0,001 µm.

Sub index2: The value of speed increment is given as a multiple of 0,01 mm/s.

9.3 Object 6010h: Preset value



Risk of injury and damage to property by an actual value jump when the Preset adjustment function is performed!

- The preset adjustment function should only be performed when the measuring system is at rest, otherwise the resulting actual value jump must be permitted in the program and application!

The Preset Function can be used to adjust the measuring system to any position value within a range of 0 to measuring length in increments –1.
The output position value is set to the parameter "Preset value" when writing to this object.

Signed32

Preset value			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 to 2^0	2^{15} to 2^8	2^{23} to 2^{16}	2^{31} to 2^{24}

Index	Sub index	Comment	Type
6010h	0	Highest sub-index supported	Unsigned8
	1	Preset value	Signed32

Sub index0: The entry at sub-index 0 contains the highest sub-index that is supported. Value = 1.

Sub index1: The entry at sub-index 1 contains the preset value for the position output.

9.4 Object 6020h: Position value

The object 6020h "Position value" defines the output of the position value for the communication objects 1800h and 1801h.

Signed32

Position value			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 to 2^0	2^{15} to 2^8	2^{23} to 2^{16}	2^{31} to 2^{24}

Index	Sub index	Comment	Type
6020h	0	Highest sub-index supported	Unsigned8
	1	Position value	Signed32

Sub index0: The entry at sub-index 0 contains the highest sub-index that is supported. Value = 1.

Sub index1: The entry at sub-index 1 contains the position value for the position output.

9.5 Object 6030h: Speed value

The object 6030h "Speed value" defines the optional output of the speed value for the communication objects 1800h and 1801h.

Signed16

Speed value	
Byte 0	Byte 1
2^7 to 2^0	2^{15} to 2^8

Index	Sub index	Comment	Type
6030h	0	Highest sub-index supported	Unsigned8
	1	Speed value	Signed16

Sub index0: The entry at sub-index 0 contains the highest sub-index that is supported. Value = 1.

Subindex1: The entry at sub-index 1 contains the speed value for the speed output.

9.6 Object 6200h: Cyclic timer

Defines the parameter "Cyclic timer". An asynchronous transmission of the position value is set, when the cyclic timer is programmed > 0. Values between 1 ms and 65535 ms can be selected. Default value = 0.

e.g.: 1 ms = 1 h
256 ms = 100 h

When the measuring system is started with the NODE START Command and the value of the cyclic timer is > 0, the 1st transmit PDO (object 1800h) transmit the measuring system position.

9.7 Object 6500h: Operating status

This object contains the operating status of the measuring system. It gives information on measuring system internal programmed parameters.

Unsigned16

Bit	Function	Bit = 0	Bit = 1
0-2	Reserved		
3	Code Sequence	increasing	decreasing
4- 15	Reserved		

9.8 Object 6501h: Measuring step, linear

For linear measuring systems object 6501h indicates the measuring step that is output by the measuring system. The measuring step is given in nm (0.001µm).
E.g.: 1µm = 00 00 03 E8 h

Unsigned32

Measuring step			
Byte 0	Byte 1	Byte 2	Byte 3
2^7 to 2^0	2^{15} to 2^8	2^{23} to 2^{16}	2^{31} to 2^{24}

Standard value: 186A0 h = 100 000 = 0.1 mm

9.9 Object 6503h: Alarms

Additionally to the emergency message, object 6503h provides further alarm messages. An alarm is set if a malfunction in the measuring system could lead to an incorrect position value. If an alarm occurs, the according bit is set to logical high until the alarm is cleared and the measuring system is able to provide an accurate position value.

Unsigned16

Bit	Function	Bit = 0	Bit = 1
0	Position error	No	Yes
1	Reserved for further use		
2	Reserved for further use		
3	Reserved for further use		
4	Reserved for further use		
5	Reserved for further use		
6	Reserved for further use		
7	Reserved for further use		
8	Reserved for further use		
9	Reserved for further use		
10	Reserved for further use		
11	Reserved for further use		
12	Manufacturer specific functions		
13	Manufacturer specific functions		
14	Manufacturer specific functions		
15	Manufacturer specific functions		

Position error

The bit is set, if no magnet is detected.

9.10 Object 6504h: Supported alarms

Object 6504h contains the information on supported alarms by the measuring system.

Unsigned16

Bit	Function	Bit = 0	Bit = 1
0	Position error	No	Yes
1	Reserved for further use		
2	Reserved for further use		
3	Reserved for further use		
4	Reserved for further use		
5	Reserved for further use		
6	Reserved for further use		
7	Reserved for further use		
8	Reserved for further use		
9	Reserved for further use		
10	Reserved for further use		
11	Reserved for further use		
12	Manufacturer specific functions		
13	Manufacturer specific functions		
14	Manufacturer specific functions		
15	Manufacturer specific functions		

9.11 Object 6505h: Warnings

This object is not supported.
By read access the value is always "0".

9.12 Object 6506h: Supported warnings

This object is not supported.
By read access the value is always "0".

9.13 Object 6507h: Profile and software version

This object contains in the 1st 16 bits the profile version which is implemented in the measuring system. It is combined to a revision number and an index.

e.g.:	Profile version:	3.2	
	Binary code:	0000 0011	0000 0010
	Hexadecimal:	3	2

The 2nd 16 bits contain the software version which is implemented in the measuring system. Only the last 4 digits are available.

e.g.:	Software version:	01.04	
	Binary code:	0000 0001	0000 0010
	Hexadecimal:	01	04

The complete software version is contained in object 100Ah, see page 45.

Unsigned32

Profile version		Software version	
Byte 0	Byte 1	Byte 2	Byte 3
2^7 to 2^0	2^{15} to 2^8	2^7 to 2^0	2^{15} to 2^8

9.14 Object 6508h: Operating time

This object is not supported.
The operating time function is not used the operating time value is set to the maximum value (FF FF FF FF h).

9.15 Object 650Ah: Modul identification

The sub-indices 1- 3 contains information manufacturer specific settings.

9.16 Object 650Bh: Serial number

This object contains the serial number of the device. It is equivalent to object 1018h sub index 4.

9.17 Object 650Ch: Offsetwert

This object contains the offset value, which is calculated by preset function. The offset value will be save and can be read from the measure system.

Index	Sub index	Comment	Type
650Ch	0	Highest sub-index supported	Unsigned8
	1	Offset value	Signed32



10 Emergency Message

Emergency messages are triggered by the occurrence of a device internal malfunction and are transmitted from the concerned application device to the other devices with highest priority.

Emergency Message								
Byte	0	1	2	3	4	5	6	7
Contents	Emergency Error Code Object 1003h, Byte 0-1		Error Register Object 1001h	0	0	0	0	0

COB-Identifier = 080h + Node-ID

If the measuring system detects an internal error, an emergency message will be transmitted with the error code of object 1003h (pre-defined error field) and the error register object 1001h. Additionally to the emergency object the according bit in the Alarm object 6503h is set.

If the error disappears, the measuring system transmits an emergency message with error code "0" (reset error / no error) and error register "0".

11 Transmission of the measuring system position value

Before the measuring system position can be transferred the measuring system has to be started with the “Node Start” command.

Node-Start Protocol

COB-Identifier = 0	
Byte 0	Byte 1
1	Node-ID

Node Start command with the Node-ID of the measuring system (slave) starts only this device.

Node Start command with **Node-ID = 0** starts all slaves connected to the bus.

After the Node Start command the measuring system transmit the position value one time with the COB-ID of object 1800h.

Now the measuring system position value can be transmitted in different ways:

1. Asynchronous Transmission

The 1st transmit PDO (object 1800h) transmit the position value of the measuring system in time intervals. The cyclic time is defined by the value of the cyclic timer (object 6200h or 1800/5). This transmission starts automatically after the Node Start command and the value of the cyclic timer is > 0.

The default value of the COB-ID is 180h + Node-ID.

Object	Function Code	COB-ID	Index Communication Parameter
PDO1 (tx)	0011bin	181h – 1FFh	1800h

In order to stop the transmission of the measuring system position temporarily, the output can be interrupted by timer value = 0, in object 6200h (or 1800/5).

2. Synchronous Transmission

The 2nd transmit PDO (object 1801h) transmit the position value of the measuring system on request (remote / sync), non-recurrent triggering.

- The measuring system receives a remote frame with the COB-ID (default value 280h + Node-ID)

Object	Function Code	COB-ID	Index Communication Parameter
PDO2 (tx)	0101bin	281h – 2FFh	1801h

- The measuring system receives a sync telegram with the COB-ID (default value 080h) defined in object 1005h. All slaves with this SYNC-COB-ID will transmit the position value.

Object	Function Code	COB-ID	Index Communication Parameter
SYNC	0001bin	80h	1005

12 Causes of faults and remedies

12.1 SDO Error codes

In the case of an error (SDO response CCD = 0x80) the data field contains a 4-byte error code. By the measuring system the following error codes are supported:

Error code	Meaning	Remedy
0x0600 0006	EEPROM error	Possibly shut-off measuring system voltage then switch on again. If the error recurs despite this measure, the measuring system must be replaced.
0x0601 0000	Unsupported access to an object	Check which attribute for the corresponding object is valid: - rw: read- and write access - wo: write only access - ro: read only access - Const: read only access
0x0609 0011	Sub-index does not exist	Check which sub-indices the corresponding object supports.
0x0800 0000	General error	Wrong signature written when storing the parameters, see Object 1010h: Store parameters, page 46.

Table 10: SDO Error codes

12.2 Emergency Error codes

Emergency objects are triggered by the occurrence of a device internal error situation, transmission format see chapter "Emergency Message", page 57.

The error indication is carried out about the objects

- Error register 0x1001, page 43 and
- Pre-defined error field 0x1003, page 44

12.2.1 Object 1001h: Error register

The error register displays bit coded the error state of the measuring system. Also several errors at the same time can be displayed by a set bit. The error code of the error occurred last is stored in object 0x1003, sub-index 1, the number of errors in sub-index 0. An error is signaled at the moment of the occurrence by an EMCY-message. By reading of the object 1001h the error stored last in object 0x1003, sub-index 0, is cleared. Each further read request clears a further error from the list. With the clearing of the last error the error register is set back and an EMCY-message with error code "0x000" is transferred.

Bit	Meaning
0	generic error
1	0
2	0
3	0
4	communication
5	device profile specific
6	0
7	sensor

12.2.2 Object 1003h: Pre-defined Error field, bits 0-15

About the Emergency object only the error occurred last is indicated. For each EMCY-message which could be deleted an EMCY-report with error code "0x0000" is transmitted. The result can be taken from object 0x1003. If no more error is present, the error register indicates also no more error.

The error list in object 0x1003 can be deleted in different ways:

1. Writing a “0” to sub-index 0 in object 0x1003
2. Execution of the NMT-service “Reset Communication”, command 0x82
3. Reading the object 0x1001, after the last error was deleted

Error code	Meaning	Remedy
0x0000	reset error / no error	-
0x8100	Communication errors, which are triggered by the CAN-controller.	- Reset node with command 0x81, after that start the node again with command 0x01. - Switch off; switch on again the supply voltage of the measuring system.

Table 11: Emergency Error codes

12.3 Alarm messages

About the object 6503h additionally to the EMCY-message further alarm messages are output. The corresponding error bit is deleted, if the error is present no more.

Error	Cause	Remedy
Bit 0 = 1, Position error	Failure of scanning elements in the measuring system Measuring system has detected no magnet.	Slide magnet into measuring range.
Bit 12 = 1, EEPROM error	Memory area in internal EEPROM defective	Possibly shut-off measuring system voltage then switch on again. If the error recurs despite this measure, the measuring system must be replaced.
Bit 13 = 1, Parameter error	Programmed parameter out of range.	Check min. and max. values of each parameter.

12.4 Other faults

Fault	Cause	Remedy
Position skips of the measuring system	Strong vibrations	Vibrations, impacts and shocks, e.g. on presses, are dampened with "shock modules". If the error recurs despite these measures, the measuring system must be replaced.
	Electrical faults EMC	Perhaps isolated flanges and couplings made of plastic help against electrical faults, as well as cables with twisted pair wires for data and supply. Shielding and wire routing must be performed according to the respective field-bus system construction guidelines.
	Extreme axial and radial load on the shaft or a defective scanning unit.	Couplings prevent mechanical stress on the shaft. If the error still occurs despite these measures, the measuring system must be replaced.



HYDAC ELECTRONIC GMBH
Hauptstr.27
D-66128 Saarbrücken
Germany

Web: www.hydac.com
E-Mail: electronic@hydac.com
Tel.: +49 (0)6897 509-01
Fax.: +49 (0)6897 509-1726

HYDAC Service
For enquiries about repairs or alterations, please contact HYDAC Service.

HYDAC SERVICE GMBH
Hauptstr.27
D-66128 Saarbrücken
Germany

Tel.: +49 (0)6897 509-1936
Fax.: +49 (0)6897 509-1933

Note

The information in this manual relates to the operating conditions and applications described. For applications and operating conditions not described, please contact the relevant technical department.

If you have any questions, suggestions, or encounter any problems of a technical nature, please contact your Hydac representative.

Subject to technical modifications.